AÑO III · N.º 100

MICROHOBBY

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES SINCLAIR

SEMANAL

AÑO III · N.º 100

135 PTS.
Canarias 140 ptas.

MICROPANORAMA

EPIX: el monstruo del software americano

NUEVO

ASTERIX Y EL CALDERO MAGICO, EL COMIC EN TU ORDENADOR

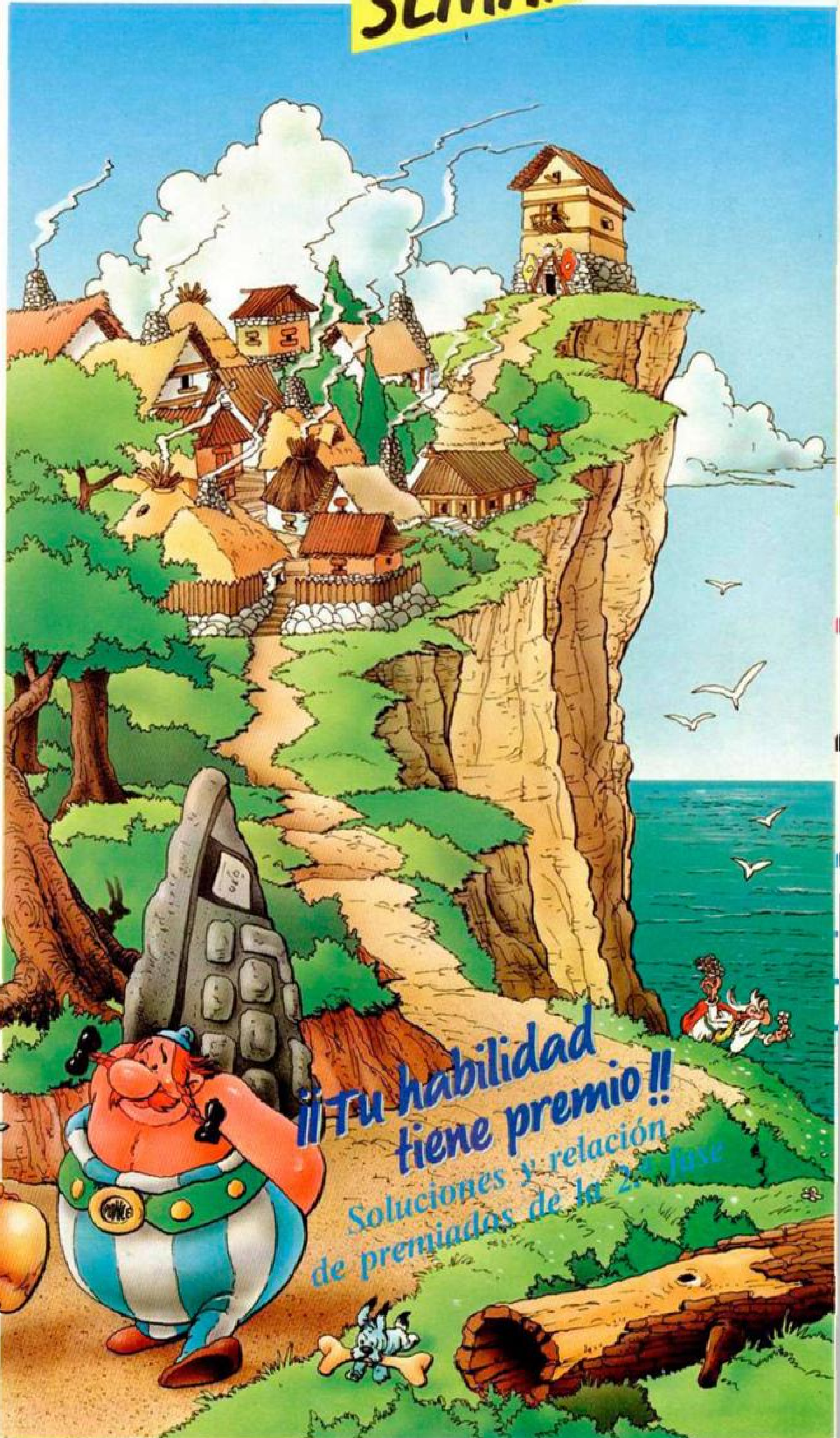
¡¡incluido mapa!!

PROGRAMACION

Movimiento de los personajes secundarios

MICROMANIA

Las claves para llegar al final de "El Zorro"



HOBBY PRESS

¡¡¡No te quedes colgado!!!

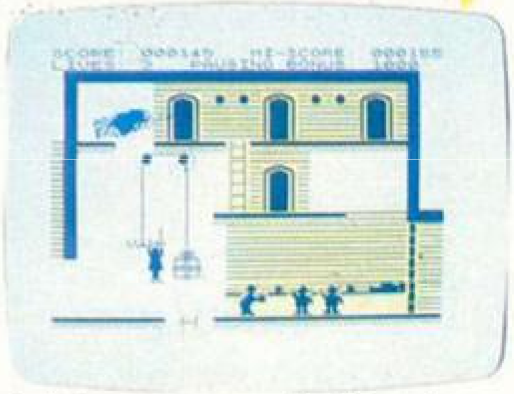


***C**ompleta tu colección de Microhobby. Solicita hoy mismo los ejemplares que te faltan. Ya hay algunos números agotados.*
(Rellena el cupón que aparece cosido en las páginas de esta revista).

MICROHOBBY ESTA SEMANA

AÑO III. N.º 100. 28 del X al 3 del XI de 1986.
135 ptas. Canarias, Ceuta y Melilla: 130 ptas.
Sobretasa aérea para Canarias: 10 ptas.





Todos los secretos acerca de «El Zorro» los encontrarás en la pág. 15.

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo
Domingo Gómez

Asesor Editorial
Gabriel Nieto

Diseño
Jaime González,
Cristina Gómez

Redacción
Amalio Gómez, Pedro Pérez,
Jesús Alonso

Secretaría Redacción
Carmen Santamaría

Colaboradores
Primitivo de Francisco,
Rafael Prades,
Miguel Sepúlveda, Sergio Martínez
J. M. Lazo
Paco Martín

Corresponsal en Londres
Alan Heap

Fotografía
Carlos Candel
Chema Sacristán

Portada
José María Ponce

Dibujos
Teo Mójica, F. L. Frontán,
J. M. López Moreno,
J. Igual, J. A. Calvo,
Lóriga, J. Olivares

Edita
HOBBY PRESS, S. A.

Presidente
María Andrino

Consejero Delegado
José I. Gómez-Centurión

Jefe de Producción
Carlos Peropadre

Publicidad
Mar Lumbreras

Secretaría de Dirección
Pilar Aristizábal

Suscripciones
M.ª Rosa González
M.ª del Mar Calzada

Redacción, Administración y Publicidad
Ctra. de Irún Km 12,400
28049 Madrid
Tél: 734 70 12
Télex: 49480 HOPR

Pedidos y Suscripciones
Tél: 734 65 00

Dto. Circulación
Paulino Blanco

Distribución
Coedis, S. A. Valencia, 245
Barcelona

Imprime
Rotedic, S. A. Ctra. de Irún,
km 12,450 (MADRID)

Fotocomposición
Novocomp, S.A.
Nicolás Morales, 38-40

Fotomecánica
Grof
Ezequiel Solana, 16

Depósito Legal
M-36.598-1984

Representante para Argentina, Chile, Uruguay y Paraguay, Cía. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64. 1209 BUENOS AIRES (Argentina)

Solicitado control
OJD

MICROHOBBY NUMEROS ATRASADOS

Queremos poner en conocimiento de nuestros lectores que para conseguir números atrasados de MICROHOBBY SEMANAL, no tienen más que escribirnos indicándonos en sus cartas el número deseado y la forma de pago elegida de entre las tres modalidades que explicamos a continuación.

Una vez tramitado esto, recibirá en su casa el número solicitado al precio de 95 ptas. + 6 de IVA hasta el n.º 36, a 126 ptas. + 8 de IVA hasta el n.º 60 y a 135 ptas. desde el n.º 60 en adelante.



FORMAS DE PAGO

- Enviando talón bancario nominativo a Hobby Press, S. A. al apartado de Correos 54062 de Madrid.
- Mediante Giro Postal, indicando número y fecha del mismo.
- Con Tarjeta de Crédito (VISA o MASTER CHARGE), haciendo constar su número y fecha de caducidad.

Versión de Ocean de la película Highlanders

La versión para ordenador del film de reciente estreno en España y en todo el mundo, Highlanders, está servida. Esta gran película de aventuras, protagonizada por Christopher Lambert y Sean Connery, narra las peripecias y enfrentamientos que tienen lugar entre un grupo de inmortales que habitan en la Tierra.

La historia tiene sus comienzos en las escarpadas costas escocesas, allá por la remota Edad Media, y se desarrolla en las épocas sucesivas hasta llegar a la actualidad en la ciudad de Nueva York.

La película, al igual que el juego, posee como elementos principales a las espadas. Estas son las armas favoritas de los inmortales y no se separan de ellas ni un instante. Highlanders, el programa para Spectrum, Commodore y Amstrad, ha tomado el argumento de la película y ha llevado a cabo un arcade en el que representamos el papel de uno de estos inmortales, que se ve obligado a luchar con el resto; tan sólo uno puede sobrevivir.



A partir de aquí se desata el argumento y comienza el desarrollo de la acción. Los inmortales se perseguirán a través de los siglos con el único objetivo de sobrevivir a los demás.

De entre los inmortales destacan dos sobre los demás: Macleod y Kurgan, el bueno y el malo respectivamente. Entre ellos se desata el combate final y en él se puede decidir el futuro de la tierra y sus pobladores. Si vence Macleod triunfará el bien, la paz reinará eternamente y a la vez este inmortal recobrará su condición de persona normal. Si por el contrario es Kurgan quien resulta vencedor una era de tinieblas y sombras se apoderará del mundo.

Ahora Ocean pone en tus manos la posibilidad de decidir el destino de los hombres. Si te decides a hacerlo aún tendrás que esperar algunas semanas, puesto que el programa aún no está a la venta en nuestro país.

The Code Machine, ensamblador/desensamblador editado por Discovery

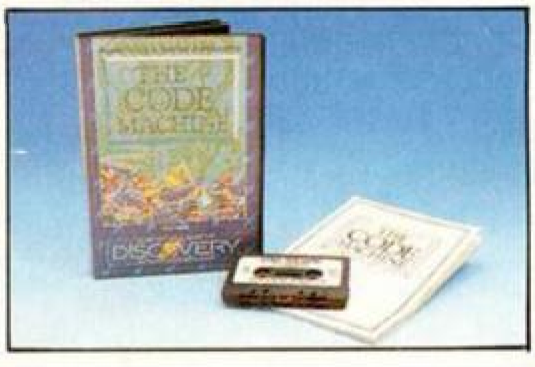
Code Machine es un paquete de desarrollo para programación en código máquina que consiste en un editor/ensamblador y en un monitor/desensamblador. Ambos programas están incluidos en la misma cinta y están escritos enteramente en código máquina.

El editor/ensamblador *(«Editas»)*, permite introducir y editar programas fuente en código máquina, producir programas objeto, imprimir los listados de los programas, así como cargar y grabar los mismos.

El Monitor/Desensamblador *(«Monitor»)*, ofrece todos los comandos necesarios para comprobar y depurar los programas escritos en código máquina, del mismo modo que facilita el estudio del funcionamiento interno del Spectrum.

Ambos programas son muy rápidos y sencillos de manejar, lo que hace que Code Machine sea una excelente ayuda para los que comienzan a tener una primera toma de contacto con el lenguaje máquina.

Code Machine no es un manual para aprender a programar en este complicado lenguaje, pero sí puede ser de mucha utilidad a la hora de depurar y solucionar muchos problemas que se puedan desprender de dicho aprendizaje, pues permitirá corregir sobre la marcha los errores que se vayan cometiendo.



AQUI LONDRES

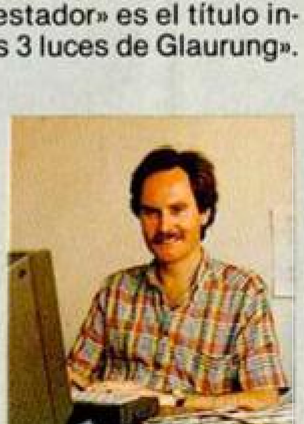
Desde el primer lanzamiento de «Bug Byte» en la feria PCW del año pasado, la variedad de juegos y las ventas de esta compañía han aumentado de manera extraordinaria.

Los primeros lanzamientos de este otoño han sido «Miami Dice», una simulacion del juego de dados americanos «Craps». Este juego se puede conseguir para el Spectrum y Commodore 64 y pronto podrá ser adquirido por los usuarios de Amstrad CPC al precio de 3 libras.

«Captain Kellh» es un nuevo juego de acción presentado por Quicksilva. En él, el jugador toma el papel de «Captain Kellh», un silencioso y diestro piloto, que entra en Obsidion II, un transportador espacial protegido por robots.

Ningún jugador conseguirá llegar al final a menos que actúe con sigilo y estrategia ya que la acción transcurre veloz y con furia. El juego se podrá conseguir para el Commodore 64 y Spectrum, tendrá un precio de 9 libras.

ERBE Software, acaba de lanzar «Conquestador» a través de Melbourne House. Es el primer producto que esta casa española publica en el Reino Unido. Por su parte, Melbourne House tiene grandes esperanzas en su éxito y confía en comercializar más software español en el futuro. «Conquestador» es el título inglés de «Las 3 luces de Glaurung».



Periféricos musicales diseñados por Cheetah

Cheetah, una de las compañías más importantes dedicadas al diseño de periféricos para Spectrum, presenta dos nuevos productos: Sound Sampler y MIDI Interface.

Sound Sampler es un periférico de tratamiento musical cuyas principales prestaciones son las siguientes: reproduce cualquier sonido previamente grabado, amplitud de onda de 17.5 Khz, salida para HIFI, ecos y reverberaciones, compatibilidad con microdrive, ejecuta el sonido desde el teclado y otras muchas opciones que permiten obtener unas grandes posibilidades sonoras y musicales del Spectrum.

MIDI Interface es otro periférico con orientaciones sonoras, pero con aplicaciones bien distintas. Su principal función es la de permitir controlar otros instrumentos musicales que puedan conectarse al ordenador a través de él. De esta forma, permite grabar melodías compuestas en un teclado *(por ejemplo)* que luego serán reproducidas automáticamente gracias a la memoria del microordenador. Puede ser utilizado en un Spectrum 48 ó 128 K.

Sus prestaciones son muy buenas, pero más si se tiene en cuenta lo reducido de su precio, alrededor de las 10.000 ptas.





20 +

CLASIFICACION	SEMANAS PREM.	TENDENCIA		SPECTRUM	AMSTRAD	COMMODORE	MSX
1	5	↑	**KUNG-FU MASTER.** US Gold	●			●
2	16	↑	**THE WAY OF THE TIGER.** Gremlin	●	●		
3	7	–	**LAS TRES LUCES DE GLAURUNG.** Erbe	●	●		
4	16	↓	**GREEN BERET.** Imagine.	●	●	●	
5	7	↑	**PHANTOMAS II.** Dinamic	●			
6	6	↑	**SUPERSERIES.** Dinamic	●			
7	5	↓	**STAINLESS STEEL.** Mikro-Gen	●	●		
8	16	–	**WORLD SERIES BASKETBALL.** Imagine	●			
9	16	–	**BATMAN.** Ocean	●	●		
10	5	↓	**PYRACURSE.** Hewson	●			
11	5	↑	**EQUINOX.** Mikro-Gen	●	●		
12	5	–	**CAULDRON II.** Palace Soft	●		●	
13	4	↑	**JACK THE NIPPER.** Gremlin	●	●		●
14	15	↓	**MOVIE.** Imagine	●	●		
15	13	↑	**PING PONG.** Imagine	●	●		
16	2	↑	**PENTAGRAM.** Ultimate	●	●		
17	15	↑	**CAMELOT WARRIORS.** Dinamic	●	●		
18	5	↓	**PHANTOMAS.** Dinamic	●			
19	15	↑	**TURBO ESPRIT.** Durell	●	●		
20	1	↑	**SPINDIZZY.** Electric Dreams	●	●		

Esta información ha sido elaborada con la colaboración de los centros de Microinformática de El Corte Inglés.

G. FREEMAN Y J. BRAZIER, DIRECTORES DE EPYX

EPYX, EL MONSTRUO DEL SOFTWARE AMERICANO

Gilbert K. Freeman y Jhon C. Brazier, presidente y vicepresidente de Epyx, una de las compañías de software más importantes del mundo, estuvieron recientemente en nuestro país. La visita se incluía dentro de la gira que dichos directivos han realizado a todos los países de Europa en los que la compañía tiene sus productos a la venta.

Aprovechamos su estancia en en España para charlar sobre algunos aspectos de la actualidad de la propia compañía y de la situación del software en Estados Unidos.

—¿Cuál es la situación actual del mercado de los ordenadores personales en USA?

—Francamente favorable. En USA existen una gran cantidad de home-computers debido a lo asequible de su precio. Prácticamente todas las familias poseen uno en su casa, siendo los jóvenes los que hacen un mayor uso de ellos. Lo bajo de su coste, —un Commodore 64 viene a valer menos de 40.000 ptas. con unidad de disco incluida—, permite que los chicos ahorren durante una temporada y acaben comprándose uno.

—¿Cuáles son las compañías punteras en la venta de ordenadores personales?

—Bueno, en realidad el mercado en los Estados Unidos es muy diferente al que tienen aquí en Europa. Allí no existe la variedad de marcas que aquí, —Sinclair, Amstrad, MSX, Commodore,...— y todas ellas de una gran calidad, sino que Commodore representa prácticamente el monopolio del mercado. Apple le sigue en el ranking, pero a una enorme distancia.



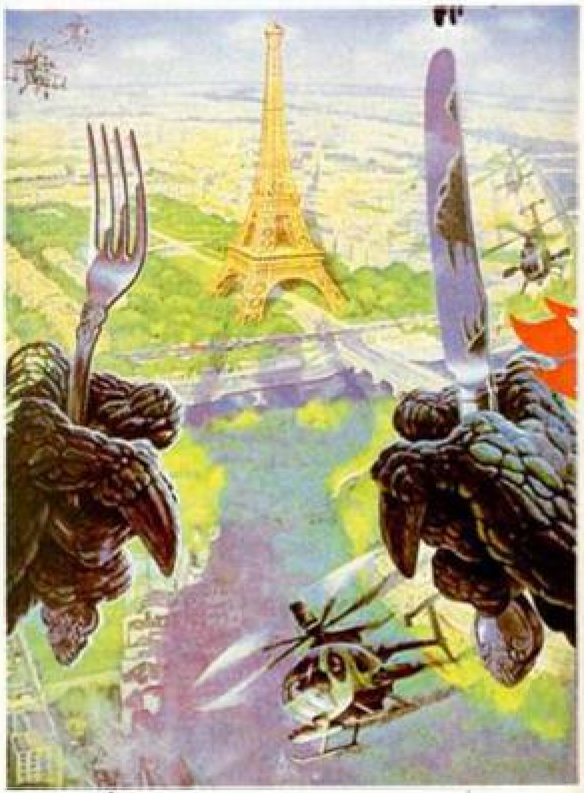
—¿Qué tal aceptación están teniendo en su país los nuevos 520 ST de Atari y el Amiga de Commodore?

—El Amiga es uno de los mejores ordenadores de cuantos existen en el mercado. Sus posibilidades gráficas son realmente maravillosas, increíbles. Parece mentira que de un monitor puedan salir semejantes imágenes. Pero aunque Commodore lo diseñó con la intención de darle una orientación hacia los juegos, el precio con el que salió al mercado fue demasiado alto. Han intentado abaratarlo, pero evidentemente existe un tope, el cual sigue siendo demasiado alto para sus posibles compradores. Por eso ahora Commodore está pensando en orientarlo hacia un campo más profesional, pero cuenta con el inconveniente de que aún no dispone del software necesario. Es una lástima porque el Amiga es la máquina perfecta para la creación de gráficos.

El Atari ST se está vendiendo, pero no con tanto éxito como aquí en Europa.

—En cuanto al software, ¿cuáles son, en la actualidad las compañías más importantes?

—Existen muchas compañías dedicadas al software, pero a parte de nosotros existen otras dos grandes casas, Activision y Electronic Arts.



«Movie Monster», la emoción está servida.

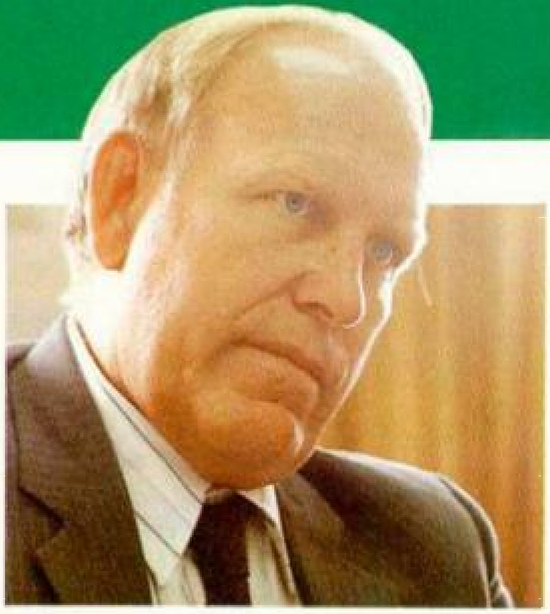
—¿Cuándo y cómo empezó Epyx a realizar programas?

—Empezamos hace ya ocho años, en 1978, con una aventura llamada Temple of Arcshire. Este programa se hizo muy famoso y en poco tiempo empezamos a desarrollar otros programas de acción. Pero fue en 1984 cuando lanzamos uno de los mayores booms de la compañía: Olimpics. A partir de este momento descubrimos que los juegos deportivos se prestaban muy bien para ser llevados al ordenador y obtener una gran adicción por parte de los usuarios. Por eso empezamos a hacer programas como Summer Games o Winter Games, los cuales han llegado a ser verdaderos éxitos en todo el mundo.

—¿Cuál cree que es la clave del éxito de sus programas y cuántas copias deben venderse en Estados Unidos para que un juego pueda considerarse como un hit?

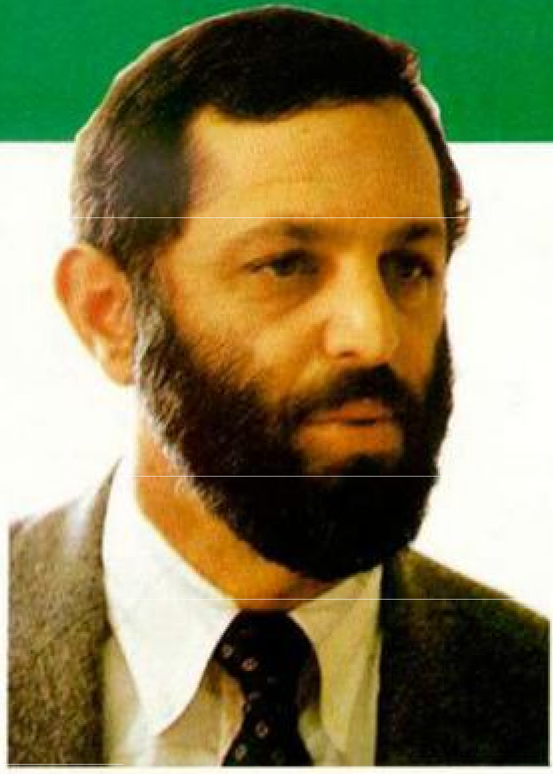
—Creemos que el punto que hace que nuestros juegos tengan tanta aceptación es el gran nivel de sus gráficos. Ahí es donde ponemos todo nuestro empeño. Queremos que cuando el usuario cargue uno de nuestros programas se quede auténticamente admirado con la calidad de las pantallas. Luego, por supuesto, también es importante la elección de un tema interesante, pero pensamos que unos buenos diseños hacen bueno a cualquier argumento.

En cuanto a las copias que solemos vender de nuestros éxitos vienen a rondar las 250.000, que fue lo que vendimos



Gilbert K. Freeman, presidente de Epix.





A la reunión también asistió el vicepresidente de la compañía, John C. Brazier.

del Summer Games. Esto sin contar las ventas en Europa.

—¿Qué opinan del estado del mercado del software en Europa?

—Es excitante. Llevamos muy poco tiempo trayendo nuestros programas a los europeos, pero estamos quedando francamente sorprendidos del dinamismo y movimiento que está tomando este mercado en países como Gran Bretaña, Alemania o España. Aunque evidentemente la proporción de ventas es pequeña aún con respecto a nuestro país, con un potencial de usuarios mucho mayor, estamos asombrados de la excitación que existe actualmente en Europa.

—¿Creen que la piratería influye en el nivel de ventas de Europa?

—Por supuesto, pero en Estados Unidos ocurre exactamente igual, o peor. La piratería alcanzan allí unos niveles exagerados. Un ejemplo. Nosotros sacamos un programa al mercado y a las dos semanas existe un teléfono a tu disposición al cual, si posees un Modem puedes llamar y obtener una copia pirata al instante.

Pero esto es algo contra lo que no se puede luchar. Forma parte del negocio. Continuamente cambiamos los sistemas de protección, pero siempre aparece alguien que lo inutiliza. Podríamos realizar sistemas mucho más complicados, pero eso podría hacer que el programa no funcionase, por lo que preferimos resignarnos y aceptar a la piratería como una lacra que hay que arrastrar. Esto, evidentemente, viene a repercutir directamente sobre el usuario, que ve como cada vez que adquiere un programa original tiene que desembolsar hasta 34 dólares (4.500 ptas.).

—Por último, la pregunta ineludible, ¿cuáles serán sus próximos lanzamientos?

—Tenemos preparados cuatro nuevos programas. Bueno, nosotros en realidad, y por las condiciones de nuestro mercado, sólo trabajamos para Commodore, Atari, IBM y este tipo de máquinas, pero aquí en Europa tenemos equipos de programadores encargados de llevar a cabo las versiones correspondientes para los ordenadores europeos como suelen ser Amstrad, MSX y Spectrum.

Los nuevos títulos serán Movie Moster, un arcade en el que asumimos el papel de un enorme monstruo prehistórico que ataca las ciudades más importantes del mundo y al que todos intentan destruir; Supercicle, simulación de una carrera de motos; Championship Wrestling, juego de lucha libre y, por último, un excelente programa al estilo de Summer Games que se llamará World Games y que constará de varias pruebas deportivas, pero en esta ocasión muy particulares. En ella se incluirán los deportes más extraños que se realizan en los países de todo el mundo, tales como salto de barril de Gran Bretaña, pruebas de skí de Austria, los saltos al mar de Acapulco...

—...,toros de España...

Bueno, no tiene toros. Hay muchas diferencias entre el mercado europeo y el americano. Las corridas de toros tienden a ser algo violento y nosotros no podemos sacar demasiada violencia en nuestros juegos. En Europa se hacen muchos programas que no se pueden llevar a USA; la gente no los aceptaría.

TRUCOS

INVERSION DE CARACTERES II

Oscar Luis Esteba, de Madrid, preocupado por la extensión de la rutina publicada en el n.º 85, nos ha preparado una nueva rutina que ocupa nada más y nada menos que 21 bytes, lo que supone una sensible reducción en algo más de 60 bytes.

Para utilizarla, basta con teclear el listado Basic y ejecutarlo o, con ayuda del GENS, teclear el listado ensamblador y reubicarlo en la posición donde más nos interese.

LISTADO ENSAMBLADOR

```
10         ORG  40000
20         LD   HL,16384
30         LD   DE,6144
40 NXTBYT  LD   B,8
50 NXTBIT  RR   (HL)
60         RL   A
70         DJNZ NXTBIT
80         LD   (HL),A
90         DEC  DE
100        INC  HL
110        LD   A,D
120        OR   E
130        JR   NZ, NXTBYT
140        RET
```

LISTADO BASIC

```
10 FOR A=4E4 TO 40021
20 READ B: POKE A,B
30 NEXT A
40 DATA 33,0,64,17,0,24,6,8,20
3,30,203,23,16,250,119,27,35,122
,179,32,241,201
50 RANDOMIZE USR 4E4
```

JUEGO DE CARACTERES

Desde Buenos Aires (Argentina) nos llega un pequeño programa que contiene un juego de caracteres bastante simpático.

Para utilizarlo basta con teclear el listado 1 y grabarlo en una cinta. A continuación cargaremos el cargador universal de código máquina y teclearemos el listado 2; después realizaremos un DUMP en la dirección 40000 y salvaremos el código indicando como dirección la 40000 y 207 como número de bytes.

Ya nada más nos queda cargar el programa para poder utilizar los caracteres.

Gerardo A. Lucero

LISTADO 2

LINEA	DATOS	CONTROL
1	00001C2262A21D000020	383
2	203C62A21D0000001C20	441
3	60A01F000004043C44C4	619
4	3F000000384478C03F00	562
5	000C141834D615180000	367
6	3C4444BE053800404078	695
7	4444C300001000101010	395
8	EF000004000404 5EA518	534
9	004060506050CF000000	631
10	1414141 8EF0000006C92	577
11	92929300000078444444	763
12	C7000000384E45C53800	655
13	0000784444FA41400000	635
14	3C44468D040600005C24	525
15	2424E300000010304884	567
16	38000010381010F00F00	418
17	0000444444C438000000	459
18	44446AAA110000009292	721
19	92926D000000442810A8	693
20	47000000444444BE0538	526
21	00003E4448903F000000	409

LISTADO 2

```
10 CLEAR 64599
20 LET A=64600: FOR N=15616 TO
16383: POKE A,PEEK N: LET A=A+1
: NEXT N
30 LOAD ""CODE 65120
40 POKE 23606,88: POKE 23607,2
51
```

JUEGO DE CARACTERES

a b c d e f g h i j k l m n o p
q r s t u v w x y z



¿POR QUE SERA?

A todos nosotros nos llama la atención el encontrarnos, al hacer alguna llamada del tipo RANDOMIZE USR, las cosas tan extrañas que hace nuestro ordenador. Eso le ocurre, por ejemplo, a Jesús María García de Bezana (Santander), que nos envía un par de ellos.

RANDOMIZE USR 4756, con esta llamada nos aparecerá un mensaje de todos conocido, el típico 1982 Sinclair..., pero con la gracia de no cambiar los atributos actuales de PAPER, BORDER e INK.

Si queremos hacer una pequeña broma, hacer el RANDOMIZE USR 1983, y conseguiremos que aparezca el mensaje "R Tape loading error 0:1".

Pero aquí no queda la cosa, pues Jesús Pena, de Sevilla, nos envía uno mucho más delicado, aunque muy atractivo.

RANDOMIZE USR 5050; así conseguiremos un New, pero muy atractivo, ya que pocos segundos antes de inicializarse, la pantalla se llenará de distintos atributos.

Siguiendo esta última pista, hemos encontrado otra llamada interesante, como es:

RANDOMIZE USR 5080, que al ejecutarla, curiosamente, se convierte en una línea 0 y el resto de listado que hubiera en memoria no será visible, en pantalla, fácilmente.

En este espacio también tienen cabida los trucos que nuestros lectores quieran proponer.

Para ello, no tienen más que enviarlos por correo a MICROHOBBY, Ctra. de Irún km 12,400 28049 Madrid.

ALMACEN LUNAR

Francisco CALLE CARRASCOSA

Spectrum 48 K

Por un error en el ordenador de su flamante nave intercósmica, Manolo Minglanillas ha aterrizado en la Luna, y unos piratas extraterrestres le han robado la carga.

Tras investigar dónde podían encontrarse los bultos sustraídos del compartimento de carga de la nave, ha llegado a la conclusión de que el único sitio donde pueden esconderse es en los antiguos pasadizos creados por la N.A.S.A. en el siglo XX para espiar a la R.U.S.A.

El juego consiste en reunir los 10 paquetes para poder volver a la Tierra, contando con un número determinado de centímetros cúbicos de combustible, y una cantidad ilimitada de obstáculos.

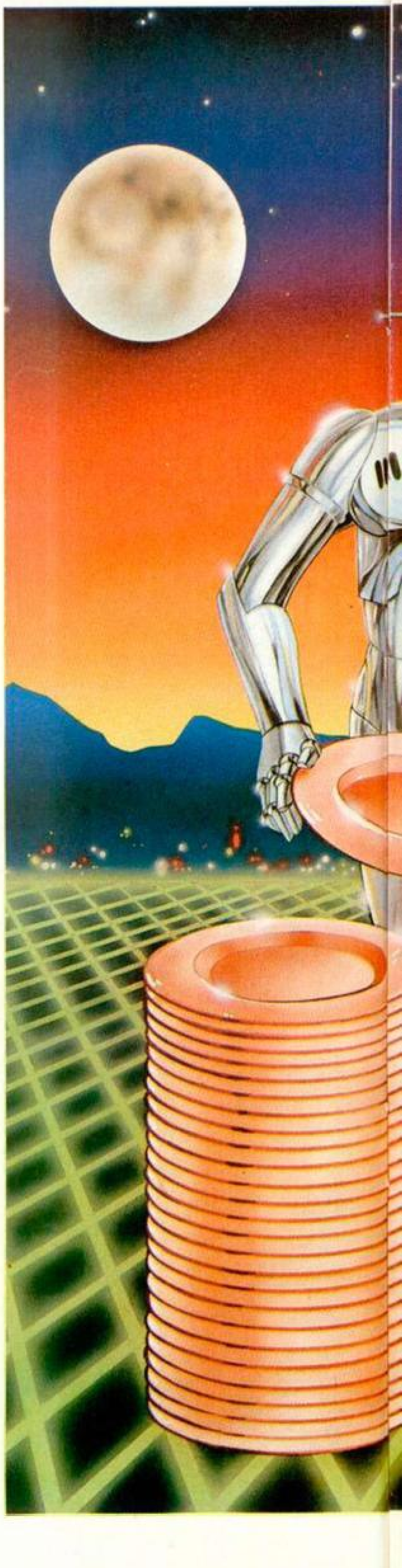
Para mover al personaje podemos utilizar Interface II o las siguientes teclas:

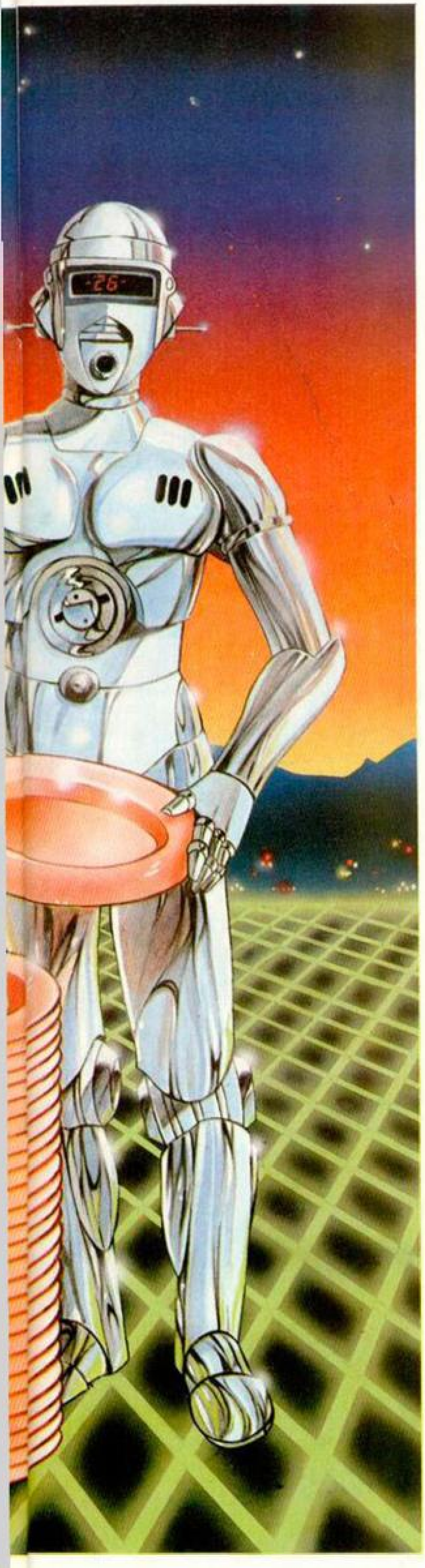
6-Izquierda	7-Derecha
8-Abajo	9-Arriba

Las pantallas del juego están distribuidas de forma laberíntica, repletas de múltiples y diversas trampas diseñadas para evitarnos llegar hasta los indispensables bultos robados, si no queremos que Manolo sea despedido en ingresar en las frondosas listas del paro, tenemos que ayudarle a conseguir llegar al final de su objetivo.

Además, y por si no os habíais dado cuenta, Manolo Minglanillas es uno de los simpáticos personajes diseñados por Dinamic en nuestro número 76.

```
   1 RUN 7600
  46 IF gas<=0 THEN GO TO 9600
  48 IF obj=10 AND pant=5 AND x=
27 THEN GO TO 9800
  49 IF ATTR (y+1,x+3)=104 OR AT
TR (y+1,x-1)=104 THEN LET obj=ob
j+1: GO SUB 9700
  50 PRINT AT y,x;a$;AT y+1,x;b$
;AT y+2,x;c$
  51 PRINT INK 7;AT yy,xx;" ": L
ET xx=xx+x1: IF ATTR (yy,xx)<>71
 AND x1=+1 THEN LET x1=-1
  52 IF ATTR (yy,xx+1)<>71 AND x
1=+1 THEN LET x1=-1
  53 IF xx>=29 THEN LET x1=-1
  54 IF xx<=2 THEN LET x1=+1
  55 IF x<1 OR x>28 OR y<1 OR y>
12 THEN LET pant=pant+(-1 AND x<
1)+(1 AND x>28)+(-5 AND y<1)+(5
AND y>12): GO SUB 2950
  56 IF ATTR (yy,xx-1)<>71 AND x
1=-1 THEN LET x1=+1
  57 PRINT INK 4;AT yy,xx;"I"
  58 IF yy=INT (y+.5) OR yy=INT
(y+1.5) OR yy=INT (y+2.5) THEN I
F INT xx=x OR INT xx=x+1 OR INT
xx=x+2 THEN GO TO 9600
  60 IF (ATTR (y+3,x)=71) AND (A
TTR (y+3,x+1)=71) AND (ATTR (y+3
,x+2)=71) THEN LET caida=1: GO T
O 95
  61 IF (ATTR (y+3,x)=67) OR (AT
TR (y+3,x+2)=67) THEN GO SUB 960
0
  90 IF INKEY$<"6" OR INKEY$>"9"
 THEN PAUSE 3: GO TO 2
  95 PRINT AT y,x;"   ";AT y+1,x
;"   ";AT y+2,x;"   ": IF caida
THEN LET y=y+.5: LET caida=0
 100 IF INKEY$="9" AND x>0 AND (
ATTR (y-1,x)=71) AND (ATTR (y-1,
x+1)=71) AND (ATTR (y-1,x+2)=71)
 THEN LET y=y-1: LET gas=gas-10:
 PRINT AT 17,15;gas;" "
 101 IF INKEY$="8" AND x<29 AND
(ATTR (y+3,x)=71) AND (ATTR (y+3
,x+1)=71) AND (ATTR (y+3,x+2)=71
) THEN LET y=y+.5
 102 IF INKEY$="6" AND (ATTR (y,
x-1)=71) AND (ATTR (y+1,x-1)=71)
 AND (ATTR (y+2,x-1)=71) THEN LE
T x=x-1
 103 IF INKEY$="7" THEN POKE 236
76,60000/256: POKE 23675,60000-2
56*PEEK 23676: LET x=x+(1 AND (A
TTR (y,x+3)=71) AND (ATTR (y+1,x
+3)=71) AND (ATTR (y+2,x+3)=71))
: GO TO 120
 104 POKE 23676,65368/256: POKE
23675,65368-256*PEEK 23676
 120 REM [illegible]
 121 IF d=1 THEN LET c$="GHI": L
ET d=2: GO TO 1
 125 IF d=2 THEN LET c$="JKL": L
ET d=3: GO TO 1
 128 IF d=3 THEN LET c$="MNO": L
ET d=4: GO TO 1
 130 IF d=4 THEN LET c$="POR": L
ET d=1: GO TO 1
 150 GO TO 1
1000 LET a$="ABC": LET b$="DEF":
 LET c$="GHI": LET x=5: LET y=12
: LET d=1: LET pant=1: LET caida
=0: LET yy=9: LET x1=+1: LET obj
=0: LET gas=5000: DIM p(45): GO
SUB 2000: GO SUB 3000
1099 PRINT AT y,x;"ABC";AT y+1,x
;"DEF";AT y+2,x;"GHI"
1100 GO TO 6
2000 BEEP .75,0: BEEP .75,7: BEE
P .75,9: BEEP .75,5: BEEP .75,0:
 BEEP .75,7: PAPER 0: INK 7: BRI
GHT 1: BORDER 0: CLS : FOR p=16
TO 21: PRINT PAPER 3;AT p,0;,: N
EXT p
2010 PLOT 11,1: DRAW 233,0: DRAW
 10,10,-PI: DRAW 0,25: DRAW -10,
10,-PI: DRAW -233,0: DRAW -10,-1
0,-PI: DRAW 0,-25: DRAW 10,-10,-
PI
2020 PLOT 13,3: DRAW 229,0: DRAW
 10,10,-PI: DRAW 0,21: DRAW -10,
10,-PI: DRAW -229,0: DRAW -10,-1
0,-PI: DRAW 0,-21: DRAW 10,-10,-
PI
2030 PRINT PAPER 5; INK 2;AT 17,
8;"GASOIL: ";gas: PRINT AT 20,8;
PAPER 5; INK 2;"OBJETOS: ";obj
2100 RETURN
2950 IF x<1 THEN LET x=28
2960 IF x>28 THEN LET x=1
2970 IF y<1 THEN LET y=12
2980 IF y>12 THEN LET y=1
3000 LET xx=RND*25+3: POKE 23675
,88: POKE 23676,236: REM [illegible]
3001 INK 4: PAPER 3: GO SUB 9500
: REM [illegible]
```





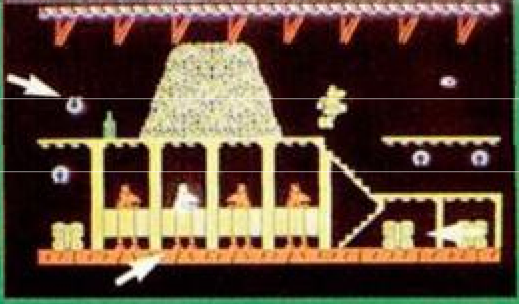
```
3002 IF pant<>1 THEN GO TO 3100
3005 FOR p=0 TO 15: PRINT AT 0,p*2; INK 1;"■■";AT 1,p*2; INK 1;"■■": NEXT p
3006 FOR p=1 TO 7: PRINT AT p*2,0; INK 1;"■■";AT p*2+1,0;"■■": NEXT p
3007 FOR p=0 TO 15: PRINT AT 15,p*2;"TU": NEXT p: PRINT AT 15,2; PAPER 0; INK 7;"    ";AT 15,26;"    "
3008 FOR p=100 TO 175-16: PLOT INK 5; PAPER 0;RND*235+16,p: NEXT p
3100 REM ■
3101 IF pant<>2 THEN GO TO 3200
3102 FOR p=0 TO 15: PRINT AT 0,p*2; INK 1;"■■";AT 1,p*2; INK 1;"■■": NEXT p
3103 FOR p=0 TO 15: PRINT AT 15,p*2;"TU": NEXT p: PRINT AT 15,16; PAPER 0; INK 7;"    "
3104 FOR p=100 TO 175-16: PLOT INK 5; PAPER 0;RND*255,p: NEXT p
3200 REM ■
3201 IF pant<>3 THEN GO TO 3300
3202 FOR p=0 TO 15: PRINT AT 0,p*2; INK 1;"■■";AT 1,p*2; INK 1;"■■": NEXT p
3203 FOR p=0 TO 15: PRINT AT 15,p*2;"TU": NEXT p: PRINT AT 15,4; PAPER 0; INK 7;"    ";AT 15,24;"    "
3204 FOR p=100 TO 175-16: PLOT INK 5; PAPER 0;RND*255,p: NEXT p
3300 REM ■
3301 IF pant<>4 THEN GO TO 3400
3302 FOR p=0 TO 15: PRINT AT 0,p*2; INK 1;"■■";AT 1,p*2; INK 1;"■■": NEXT p
3303 FOR p=0 TO 15: PRINT AT 15,p*2;"TU": NEXT p: PRINT AT 15,8; PAPER 0; INK 7;"    ";AT 15,26;"    "
3304 FOR p=100 TO 175-16: PLOT INK 5; PAPER 0;RND*255,p: NEXT p
3400 REM ■
3401 IF pant<>5 THEN GO TO 3500
3402 FOR p=0 TO 15: PRINT AT 0,p*2; INK 1;"■■";AT 1,p*2; INK 1;"■■": NEXT p
3403 FOR p=1 TO 7: PRINT AT p*2,30; INK 1;"■■";AT p*2+1,30;"■■": NEXT p
3404 FOR p=0 TO 15: PRINT AT 15,p*2;"TU": NEXT p: PRINT AT 15,12; PAPER 0; INK 7;"    "
3405 FOR p=100 TO 175-16: PLOT INK 5; PAPER 0;RND*239,p: NEXT p
3406 IF obj=10 THEN PRINT AT 11,27; FLASH 1; PAPER 0; INK 7;"   ";AT 15,27;"***"
3500 REM ■
3501 INK 1: PAPER 0: IF pant<>6 THEN GO TO 3600
3502 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD": NEXT p
3503 FOR p=3 TO 12: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3504 FOR p=1 TO 7: PRINT AT p*2,14;"AB";AT p*2+1,14;"CD": NEXT p
3505 FOR p=5 TO 10: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
3506 PRINT AT 14,16;"AB";AT 15,16;"AB"
3600 REM ■
3601 IF pant<>7 THEN GO TO 3700
3602 FOR p=3 TO 7: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3603 FOR p=10 TO 13: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3604 FOR p=1 TO 7: PRINT AT p*2,8;"AB";AT p*2+1,8;"CD": NEXT p: FOR p=3 TO 4: PRINT INK 7;AT p*2,8;"  ";AT p*2+1,8;"  ": NEXT p
3605 FOR p=4 TO 6: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
3700 REM ■
3701 IF pant<>8 THEN GO TO 3800
3702 FOR p=4 TO 11: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3703 FOR p=14 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3704 FOR p=1 TO 7: PRINT AT p*2,14;"AB";AT p*2+1,14;"CD": NEXT p
3705 PRINT AT 8,12;"AB";AT 9,12;"CD": FOR p=8 TO 10: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
3706 PRINT AT 14,30;"AB";AT 15,30;"CD"
3800 REM ■
3801 IF pant<>9 THEN GO TO 3900
3802 FOR p=0 TO 3: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p: FOR p=6 TO 12: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3803 FOR p=0 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
3804 FOR p=4 TO 6: PRINT AT p*2,16;"ABAB";AT p*2+1,16;"CDCD": NEXT p
3900 REM ■
3901 IF pant<>10 THEN GO TO 4000
3902 FOR p=3 TO 5: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p: FOR p=8 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
3903 FOR p=1 TO 7: PRINT AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p
3904 FOR p=0 TO 11: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
4000 REM ■
4001 IF pant<>11 THEN GO TO 4100
4002 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD": NEXT p
4003 FOR p=0 TO 2: PRINT AT p*2,14;"AB";AT p*2+1,14;"CD": NEXT p
4004 FOR p=0 TO 2: PRINT AT 8,p*2;"CD";AT 9,p*2;"CD": NEXT p: FOR p=7 TO 10: PRINT AT 4,p*2;"AB";AT 5,p*2;"CD": NEXT p
4005 FOR p=5 TO 8: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
4006 IF p(pant)=0 THEN PRINT AT 2,16; PAPER 5; INK 0;"IJ";AT 3,16;"KL"
4100 REM ■
4101 IF pant<>12 THEN GO TO 4200
4102 FOR p=0 TO 4: PRINT AT p*2,8;"AB";AT p*2+1,8;"CD": NEXT p
4103 FOR p=10 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
4104 FOR p=3 TO 6: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p
4200 REM ■
4201 IF pant<>13 THEN GO TO 4300
4202 FOR p=0 TO 7: PRINT AT p*2,14;"AB";AT p*2+1,14;"CD": NEXT p
4203 FOR p=0 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
4204 FOR p=4 TO 6: PRINT AT 4,p*2;"AB";AT 5,p*2;"CD": NEXT p: FOR p=8 TO 11: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p
4205 PRINT AT 0,30;"AB";AT 1,30;"CD"
4300 REM ■
4301 IF pant<>14 THEN GO TO 4400
4302 FOR p=0 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD";AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
4303 FOR p=0 TO 4: PRINT AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p
4304 FOR p=5 TO 8: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p
4400 REM ■
4401 IF pant<>15 THEN GO TO 4500
4402 FOR p=0 TO 11: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
4403 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD";AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p: PRINT INK 7;AT 10,0;"  ";AT 11,0;"  ";AT 12,0;"  ";AT 13,0;"  "
4404 PRINT AT 14,0;"AB";AT 15,0;"CD"
4405 FOR p=0 TO 5: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p: PRINT AT 6,10;"AB";AT 7,10;"CD"
4406 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 6,4;"IJ";AT 7,4;"KL"
4500 REM ■
4501 IF pant<>16 THEN GO TO 4600
4502 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD";AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p: PRINT INK 7;AT 0,30;"  ";AT 1,30;"  ";AT 2,30;"  ";AT 3,30;"  "
4503 FOR p=0 TO 3: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p: FOR p=0 TO 1: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
4600 REM ■
4601 IF pant<>17 THEN GO TO 4700
4602 FOR p=2 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD": NEXT p
4603 FOR p=0 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
4604 FOR p=0 TO 2: PRINT AT p*2,20;"AB";AT p*2+1,20;"CD": NEXT p
4605 FOR p=4 TO 6: PRINT AT 6,p*2;"AB";AT 7,p*2;"CD": NEXT p
4700 REM ■
4701 IF pant<>18 THEN GO TO 4800
4702 FOR p=0 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p
4703 FOR p=1 TO 4: PRINT AT p*2,6;"AB";AT p*2+1,6;"CD";AT p*2,22;"AB";AT p*2+1,22;"CD": NEXT p
4704 FOR p=6 TO 8: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
4705 PRINT AT 14,0;"AB";AT 15,0;"CD";AT 14,30;"AB";AT 15,30;"CD"
4706 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 8,14;"IJ";AT 9,14;"KL"
4800 REM ■
4801 IF pant<>19 THEN GO TO 4900
4802 FOR p=0 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD";AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
4803 FOR p=0 TO 7: PRINT AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p
4804 FOR p=11 TO 14: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p
4805 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 12,28;"IJ";AT 13,28;"KL"
4900 REM ■
4901 IF pant<>20 THEN GO TO 5000
4902 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD";AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p
4903 FOR p=3 TO 6: PRINT AT 6,p*2;"AB";AT 7,p*2;"CD": NEXT p
4904 FOR p=9 TO 12: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
5000 REM ■
5001 IF pant<>21 THEN GO TO 5100
5002 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD": NEXT p
5003 FOR p=0 TO 2: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p: PRINT AT 0,30;"AB";AT 1,30;"CD"
5004 FOR p=7 TO 8: PRINT AT 4,p*2;"AB";AT 5,p*2;"CD": NEXT p: PRINT AT 14,30;"AB";AT 15,30;"CD"
5005 FOR p=14 TO 15: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p
5100 REM ■
5101 IF pant<>22 THEN GO TO 5200
5102 FOR p=0 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD": NEXT p: PRINT AT 14,0;"AB";AT 15,0;"CD"
5103 FOR p=0 TO 7: PRINT AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p
5104 FOR p=0 TO 2: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p
5105 FOR p=6 TO 8: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
5200 REM ■
5201 IF pant<>23 THEN GO TO 5300
5202 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD";AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p
5203 FOR p=0 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
5204 FOR p=2 TO 4: PRINT AT p*2,16;"AB";AT p*2+1,16;"CD": NEXT p
5300 REM ■
5301 IF pant<>24 THEN GO TO 5400
5302 FOR p=0 TO 15: PRINT AT 0,p*2;"AB";AT 1,p*2;"CD";AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p: PRINT INK 7;AT 14,4;"    ";AT 15,4;"    "
5303 FOR p=0 TO 7: PRINT AT p*2,0;"AB";AT p*2+1,0;"CD": NEXT p
5400 REM ■
5401 IF pant<>25 THEN GO TO 5500
5402 FOR p=0 TO 7: PRINT AT p*2,30;"AB";AT p*2+1,30;"CD": NEXT p: PRINT AT 0,0;"AB";AT 1,0;"CD"
5403 FOR p=0 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD"
```

```
5404 FOR p=11 TO 13: PRINT AT 4,p*2;"AB";AT 5,p*2;"CD": NEXT p
5405 FOR p=7 TO 8: PRINT AT 8,p*2;"AB";AT 9,p*2;"CD": NEXT p: FOR p=3 TO 5: PRINT AT 10,p*2;"AB";AT 11,p*2;"CD": NEXT p
5406 FOR p=0 TO 15: PRINT AT 14,p*2;"AB";AT 15,p*2;"CD": NEXT p
5407 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 12,20;"IJ";AT 13,20;"KL"
5500 PAPER 0: INK 2: REM
5501 IF pant<>26 THEN GO TO 5600
5502 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH";AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
5503 FOR p=5 TO 8: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p: PRINT AT 7,12; PAPER 0; INK 3;"[+]"
5504 FOR p=10 TO 12: PRINT AT 10,p*2;"EF";AT 11,p*2;"GH": NEXT p
5600 REM
5601 IF pant<>27 THEN GO TO 5700
5602 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH": NEXT p: PRINT AT 0,30;"EF";AT 1,30;"GH"
5603 FOR p=0 TO 15: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p: PRINT AT 6,28;"EFEF";AT 7,28;"GHGH"
5604 PRINT INK 3;AT 7,12;"[+]"
5605 PRINT AT 14,30;"EF";AT 15,30;"GH"
5700 REM
5701 IF pant<>28 THEN GO TO 5800
5702 FOR p=0 TO 15: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH";AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT PAPER 0; INK 7;AT 14,26;"    ";AT 15,26;"    "
5703 FOR p=0 TO 2: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p: FOR p=0 TO 5: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p
5800 REM
5801 IF pant<>29 THEN GO TO 5900
5802 FOR p=0 TO 15: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH": NEXT p: PRINT PAPER 0; INK 7;AT 0,4;"    ";AT 1,4;"    "
5803 FOR p=9 TO 15: PRINT AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT AT 14,0;"EF";AT 15,0;"GH": PRINT AT 13,20; INK 3;"[+]"
5804 FOR p=5 TO 7: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p
5900 REM
5901 IF pant<>30 THEN GO TO 6000
5902 FOR p=0 TO 15: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH";AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT AT 14,26; INK 7;"    ";AT 15,26;"    "
5903 FOR p=3 TO 5: PRINT AT 12,p*2;"EF";AT 13,p*2;"GH": NEXT p
5904 FOR p=0 TO 7: PRINT AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
5905 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 12,15;"IJ";AT 13,15;"KL"
6000 REM
6001 IF pant<>31 THEN GO TO 6100
6002 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH";AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
6003 FOR p=10 TO 15: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p
6004 FOR p=1 TO 4: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p: PRINT AT 6,8;"EF";AT 7,8;"GH"
6005 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 6,2;"IJ";AT 7,2;"KL"
6100 REM
6101 IF pant<>32 THEN GO TO 6200
6102 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH";AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
6103 FOR p=4 TO 6: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p
6104 FOR p=9 TO 11: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p
6200 REM
6201 IF pant<>33 THEN GO TO 6300
6202 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH";AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
6203 FOR p=0 TO 12: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH": NEXT p
6204 FOR p=8 TO 12: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p
6300 REM
6301 IF pant<>34 THEN GO TO 6400
6302 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH": NEXT p: PRINT AT 14,30;"EF";AT 15,30;"GH"
6303 FOR p=9 TO 15: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH": NEXT p: PRINT AT 10,2;"EFEF";AT 11,2;"GHGH"
6304 FOR p=5 TO 8: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p: FOR p=10 TO 12: PRINT AT 10,p*2;"EF";AT 11,p*2;"GH": NEXT p
6400 REM
6401 IF pant<>35 THEN GO TO 6500
6402 FOR p=0 TO 7: PRINT AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p: PRINT AT 14,0;"EF";AT 15,0;"GH"
6403 FOR p=0 TO 12: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH": NEXT p: PRINT AT 6,28;"EF";AT 7,28;"GH"
6404 FOR p=4 TO 8: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p: PRINT INK 3;AT 7,12;"[+]"
6500 REM
6501 IF pant<>36 THEN GO TO 6600
6502 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH": NEXT p: PRINT AT 0,30;"EF";AT 1,30;"GH";AT 14,30;"EF";AT 15,30;"GH"
6503 FOR p=10 TO 12: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p
6504 FOR p=4 TO 7: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p
6600 REM
6601 IF pant<>37 THEN GO TO 6700
6602 FOR p=0 TO 15: PRINT AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,5;"[+]": PRINT AT 0,0;"EF";AT 1,0;"GH": PRINT AT 0,30;"EF";AT 1,30;"GH"
6603 FOR p=10 TO 12: PRINT AT 12,p*2;"EF";AT 13,p*2;"GH": NEXT p: PRINT AT 6,8;"EFEFEF";AT 7,8;"GHGHGH": PRINT AT 10,20;"EFEF";AT 11,20;"GHGH"
6700 REM
6701 IF pant<>38 THEN GO TO 6800
6702 FOR p=0 TO 7: PRINT AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p: PRINT AT 0,0;"EF";AT 1,0;"GH";AT 14,0;"EF";AT 15,0;"GH"
6703 FOR p=5 TO 8: PRINT AT 4,p*2;"EF";AT 5,p*2;"GH";AT 10,p*2;"EF";AT 11,p*2;"GH": NEXT p: FOR p=12 TO 14: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p
6704 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 6,28;"IJ";AT 7,28;"KL"
6800 REM
6801 IF pant<>39 THEN GO TO 6900
6802 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH";AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
6803 PRINT AT 14,0;"EF";AT 15,0;"GH": FOR p=3 TO 15: PRINT AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,8;"[+]"
6804 FOR p=1 TO 3: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p: FOR p=8 TO 11: PRINT AT 4,p*2;"EF";AT 5,p*2;"GH": NEXT p
6900 REM
6901 IF pant<>40 THEN GO TO 7000
6902 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH";AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p: PRINT AT 14,1;"EFEFEF";AT 15,1;"GHGHGH"
6903 FOR p=5 TO 9: PRINT AT 4,p*2;"EF";AT 5,p*2;"GH": NEXT p: FOR p=10 TO 12: PRINT AT 8,p*2;"EF";AT 9,p*2;"GH": NEXT p
7000 REM
7001 IF pant<>41 THEN GO TO 7100
7002 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH": NEXT p: FOR p=8 TO 12: PRINT AT 4,p*2;"EF";AT 5,p*2;"GH": NEXT p
7003 FOR p=0 TO 15: PRINT AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,8;"[+]"
7004 PRINT AT 12,15;"EFEF";AT 13,15;"GHGH": PRINT AT 0,30;"EF";AT 1,30;"GH"
7100 REM
7101 IF pant<>42 THEN GO TO 7200
7102 FOR p=0 TO 15: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH";AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,8;"[+]"
7103 FOR p=0 TO 7: PRINT AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p
7104 FOR p=5 TO 10: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p
7105 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 12,24;"IJ";AT 13,24;"KL"
7200 REM
7201 IF pant<>43 THEN GO TO 7300
7202 FOR p=0 TO 15: PRINT AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,15;"[+]"
7203 FOR p=0 TO 7: PRINT AT p*2,0;"EF";AT p*2+1,0;"GH": NEXT p: PRINT AT 12,28;"EFEF";AT 13,28;"GHGH"
7204 FOR p=8 TO 12: PRINT AT 4,p*2;"EF";AT 5,p*2;"GH": NEXT p: PRINT AT 0,30;"EF";AT 1,30;"GH"
7300 REM
7301 IF pant<>44 THEN GO TO 7400
7302 FOR p=0 TO 15: PRINT AT 0,p*2;"EF";AT 1,p*2;"GH";AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,22;"[+]"
7303 PRINT INK 7;AT 0,2;"      ";AT 1,2;"      ": PRINT AT 12,0;"EFEFEF";AT 13,0;"GHGHGH"
7304 FOR p=8 TO 12: PRINT AT 6,p*2;"EF";AT 7,p*2;"GH": NEXT p
7305 IF p(pant)=0 THEN PRINT PAPER 5; INK 0;AT 12,10;"IJ";AT 13,10;"KL"
7400 REM
7401 IF pant<>45 THEN GO TO 7500
7402 FOR p=0 TO 15: PRINT AT 14,p*2;"EF";AT 15,p*2;"GH": NEXT p: PRINT INK 3;AT 13,20;"[+]"
7403 FOR p=0 TO 7: PRINT AT p*2,30;"EF";AT p*2+1,30;"GH": NEXT p: PRINT AT 0,0;"EFEFEFEF";AT 1,0;"GHGHGHGH"
7404 FOR p=8 TO 12: PRINT AT 4,p*2;"EF";AT 5,p*2;"GH": NEXT p
7500 PAPER 0: INK 7: POKE 23675,88: POKE 23676,255: RETURN
7600 REM
7605 BORDER 0: PAPER 0: INK 7: CLS : PLOT 0,0: DRAW 255,0: DRAW 0,40: DRAW -255,0: DRAW 0,-40: PRINT AT 10,5;"Espera, "; FLASH 1;"generando GDUs": PRINT AT 20,2;"© Francisco Calle Carrascosa"
7607 POKE 23675,88: POKE 23676,236
7610 RESTORE 7620: FOR p=USR "a" TO USR "u"+7: READ y: POKE p,y: NEXT p
7620 DATA 17,124,124,238,238,124,124,17,136,62,62,119,119,62,62,136,17,124,124,238,238,124,124,17,136,62,62,119,119,62,62,136
7630 DATA 121,207,134,15,217,113,115,222,206,91,49,120,205,199,103,61,204,222,115,113,217,143,198,111,24,61,103,199,205,121,51,126
7640 DATA 255,193,173,149,169,181,131,254,255,131,181,169,149,173,193,127,254,131,181,169,149,173,193,255,127,193,173,149,169,181,131,255
7650 DATA 0,66,102,90,66,66,66,0,0,66,98,82,74,70,66,0,0,60,66,66,66,66,60,0,0,124,66,66,124,64,64,0,0,60,66,66,82,74,60,0
7660 DATA 0,124,66,66,124,68,66,0,0,60,64,60,2,66,60,0,255,191,255,119,122,254,255,255,255,175,245,238,60,175,127,255
7700 POKE 23675,96: POKE 23676,234
7710 RESTORE 7720: FOR p=USR "a" TO USR "t"+7: READ y: POKE p,y: NEXT p
7720 DATA 0,1,4,0,9,1,8,0,30,113,194,193,65,192,113,30,0,128,64,64,64,64,128,0,12,30,30,12,24,126,6,0
7730 DATA 0,170,85,0,127,62,0,115,96,176,112,32,112,40,16,0,0,0,2,3,6,11,5,2,227,231,67,0,3,3,128,195
7740 DATA 128,128,0,0,0,224,0,240,0,0,0,0,0,0,1,0,115,102,38,128,228,207,64,167,0,0,0,0,0,128,0,192
7750 DATA 0,0,0,0,0,0,0,0,243,102,8,47,32,125,0,62,0,0,0,0,0,128,0,0,1,1,2,6,3,5,2,1
7760 DATA 193,192,1,1,0,129,193,64,128,128,128,0,96,208,160,64,255,255,191,191,243,239,255,255,24,36,66,165,165,66,36,24
7800 POKE 23675,88: POKE 23676,255
7810 RESTORE 7820: FOR p=USR "a" TO USR "t"+7: READ y: POKE p,y: NEXT p
7820 DATA 0,1,2,2,2,2,1,0,120,142,67,131,130,3,142,120,0,128,32,0,144,128,16,0,6,13,14,4,14,20,8,0
7830 DATA 0,85,170,0,254,124,0,206,48,120,120,48,24,48,96,0,1,1,0,0,0,7,0,15,199,231,194,0,192,192,1,195
7840 DATA 0,0,64,48,96,208,160,64,0,0,0,0,0,1,0,3,206,102,100,1,39,243,2,229,0,0,0,0,0,0,128,0
7850 DATA 0,0,0,0,0,1,0,0,207,102,16,244,4,190,0,124,0,0,0,0,0,0,0,0,1,1,1,0,6,11,5,2
7860 DATA 193,1,192,64,128,192,193,129,192,192,32,48,96,208,160,64,0,60,64,60,2,66,60,0,24,36,66,165,165,66,36,24
7890 BEEP .75,0: BEEP .75,7: BEEP .75,9: BEEP .75,5: BEEP .75,0: BEEP .75,7
7900 RUN 1000
9500 REM
9510 FOR p=0 TO 15: PRINT AT p,0; PAPER 0; INK 7,,: NEXT p
9520 RETURN
9600 REM
9601 BEEP 1,0
9602 FOR p=7 TO 0 STEP -1: PRINT INK p;AT y,x;"ABC";AT y+1,x;"DEF";AT y+2,x;"JKL": BEEP .1,p*2: BEEP .1,p/2: NEXT p
9603 FOR p=17 TO 20: PRINT PAPER 2;AT p,2;"                            ": NEXT p
9604 PRINT INK 5;AT 18,3;"Lo siento pero has muerto.": FOR p=0 TO 200: NEXT p
9605 GO TO 9860
9700 REM
9701 IF ATTR (y+1,x+3)=104 THEN PRINT PAPER 0; INK 7;AT y+1,x+3;"  ";AT y+2,x+3;"  "
9705 IF ATTR (y+1,x-1)=104 THEN PRINT PAPER 0; INK 7;AT y+1,x-2;"  ";AT y+2,x-2;"  "
9707 LET p(pant)=1
9708 PRINT AT 20,16;obj
9709 IF obj=10 THEN PRINT FLASH 1; INK 3; PAPER 4;AT 18,20;"[illegible]";AT 19,20;"[illegible]"; PAPER 3; INK 4;AT 18,3;"[illegible]";AT 19,3;"[illegible]": FOR p=0 TO 10: BEEP .1,p: NEXT p: PRINT AT 17,25;" Ve ";AT 18,25;"a la ";AT 19,25;"pant.";AT 20,25;"nº5. "
9710 RETURN
9800 REM
9801 OVER 1: FOR p=0 TO 255 STEP 5: PLOT 0,48: DRAW 255-p,127: PLOT 0,175: DRAW 255-p,-127: PLOT 255,48: DRAW -255+p,127: PLOT 255,175: DRAW -255+p,-127: NEXT p: OVER 0
9810 OVER 1: FLASH 1: PAPER 0: INK 2
9815 FOR p=0 TO 15: PRINT AT p,0,,,: NEXT p
9820 OVER 0: FLASH 0
9830 FOR p=0 TO 255 STEP 2: OUT 254,p: NEXT p: BORDER 0
9840 PAPER 0: INK 1: CLS
9850 LET a$="   LO HAS LOGRADO MANOLO                  YA QUE TIENES LOS PAQUETES         NECESARIOS PARA EL VIAJE DE        REGRESO.                           ESPERO QUE TE HAYAS ENTRE-         TENIDO, GRACIAS."
9851 PRINT AT 8,0: FOR p=1 TO LEN a$: PRINT a$(p);: IF a$(p)<>" " THEN BEEP .01,(CODE a$(p))/3
9852 NEXT p
9853 FOR p=0 TO 100: NEXT p
9860 REM
9861 PRINT #0;"Quieres Jugar de nuevo (s/n) ?"
9862 IF INKEY$="s" THEN RUN 1000
9863 IF INKEY$="n" THEN STOP
9864 GO TO 9862
9999 SAVE "ASTRONAUTA" LINE 7600
```



SOLUCIONES A NUESTRO CONCURSO ¡¡Tu Habilidad Tiene Premio!!

Solución al pasatiempo n.º 1 de la 2.ª fase.







Solución al pasatiempo n.º 2 de la 2.ª fase.

		V	F
1.	El número de bytes por segundo se mide en «baudios».	☐	☒
2.	Cuando llega el momento, el héroe de «Camelot Warriors» se transforma en una rana.	☒	☐
3.	Mediante el uso del comando POINT, se puede dibujar en alta resolución.	☐	☒
4.	El lenguaje «Assembler» es el mismo que el «Código Máquina».	☐	☒
5.	Las hermanas Tania y Sonia intervienen, junto a Marlow, en «MOVIE».	☐	☒
6.	Pulsar el botón de Reset es exactamente lo mismo que apagar el ordenador durante un instante.	☐	☒
7.	Los juegos para Spectrum sólo pueden programarse utilizando este mismo ordenador.	☐	☒
8.	Al final del programa, Batman consigue rescatar a su amigo Robin.	☐	☒
9.	Las memorias RAM pueden ser leídas, pero es imposible escribir en ellas.	☐	☒
10.	Los programas en Código Máquina son siempre más rápidos que en BASIC.	☒	☐
11.	En el programa SABOTEUR, hay ciertas habitaciones a las que es imposible acceder.	☐	☒
12.	La instrucción de Código Máquina NOP detiene al Microprocesador hasta el siguiente ciclo de refresco.	☐	☒
13.	El espacio de memoria libre en un Spectrum de 48K es de 41528 bytes.	☐	☒
14.	Sir Clive Sinclair, aunque de origen inglés, nació en Australia.	☐	☒
15.	Los videojuegos deben salvarse en cinta con las instrucciones TRUE VIDEO e INVERSE VIDEO	☐	☒

Solución al pasatiempo n.º 3 de la 2.ª fase.

Frase:
Enhorabuena, por fin has conseguido descifrar el enigma de la frase misteriosa. Indica en el recuadro incluido en la página el número correcto de la clave. N.º clave: 1024.

Solución al pasatiempo n.º 4 de la 2.ª fase.

M	I	C	R	O	S
C	O	D	I	G	O
T	R	U	C	O	S
R	U	T	I	N	A
R	E	T	U	R	N

H	A	C	K	E	R
L	I	B	R	O	S
A	R	C	A	D	E
R	A	M	T	O	P
A	V	A	L	O	N

R	E	T	U	R	N
T	R	U	C	O	S
L	I	B	R	O	S
O	U	T	P	U	T
D	U	R	E	L	L

P	A	S	C	A	L
S	W	E	E	V	O
H	A	C	K	E	R
F	O	R	M	A	T
B	O	R	D	E	R

F	O	R	M	A	T
D	E	L	E	T	E
O	U	T	P	U	T
K	O	N	A	M	I
C	O	D	I	G	O

S	W	E	E	V	O
A	R	C	A	D	E
R	U	T	I	N	A
M	I	C	K	I	E
V	E	R	I	F	Y

A	V	A	L	O	N
P	A	S	C	A	L
M	I	C	R	O	S
V	E	R	I	F	Y
D	E	L	E	T	E

R	A	M	T	O	P
B	O	R	D	E	R
M	I	C	K	I	E
D	U	R	E	L	L
K	O	N	A	M	I

¡Enhorabuena a los ganadores! Pero si aún no has obtenido ningún premio, no te desanimes. Todavía tenéis todos la oportunidad de participar en la tercera fase y optar al gran premio final de un equipo HI-FI.

Los 50 Ganadores



- Iñaki Erauskin Zurrita (Lasarte-Guipúzcoa).
- José Ricardo Navarro Ramón (Utiel-Valencia).
- Agustín Guillermo Turiel Martínez (León).
- Miguel Angel Ballesteros García (Alicante).
- Fco. Javier Zorzano Alba (Nájera-Rioja).
- Fernando Caldera Alvarez (Mieres-Asturias).
- Albert Compte Braquest (San Cugat del Vallés-Barcelona).
- Ricardo Martínez Cantero (Madrid).
- Fco. Leandro Gutiérrez Murcia (Santander).
- José Luis Cea Claver (Los Negrales-Madrid).
- Andrés Ciruceda Esco (Barcelona).
- Fco. Javier Loeches Ambite (Madrid).
- José M.ª Santamaría Bo (Corbera-Valencia).
- Fernando García González (Soria).
- Eduardo Sánchez Salazar (Madrid).
- Benjamín Pérez Casado (Burgos).
- Carlos Salido Peracaula (Jerez de la Frontera- Cádiz)
- Elies Prunes Soler (Manresa-Barcelona).
- Víctor Manuel Ortega Royas (Torremolinos-Málaga).
- Manuel Muñoz Vidal (Betanzos-La Coruña).
- José M.ª Ferrer Almazán (Zaragoza).
- Raúl Cuevas Barrón (Sevilla).
- Francisco Cid Molins (Barcelona).
- Carlos Pons de Hita (Barcelona).
- Raúl Jiménez Parra (Madrid).
- Carlos López Samaniego (Madrid).
- Antonio Manuel Jurado Mejías (Dos Hermanas-Sevilla).
- José Castanedo Pérez (Santander).
- Javier Tabernero da Veiga (Villaviciosa de Odón-Madrid).
- Enrique Vega Valiente (San Fernando-Cádiz).
- Delfín López Remacho (Torremolinos-Málaga).
- Víctor Jesús Ventura Alvarez (Granada).
- Fco. Javier Blas Vaquero (Zaragoza).
- José Benjamín Blas Garrido (Gijón-Asturias).
- Jaime Cabeza Láinez (Jaén).
- Daniel Aguirre Molina (Jaén).
- Antonio Nieto-Márquez Venero (Madrid).
- Silvia Moya Martínez (Alicante).
- Rafael Ant.º Antón Sánchez (Elche-Alicante).
- José M.ª Sánchez Romera (Palma-Baleares).
- Cristóbal Antonio Pérez Gómez (Bailén-Jaén).
- Fernando González Fernández (Madrid).
- Antonio Fernández Alonso (Sevilla).
- Alberto Delgado Salan (León).
- Joaquín Manuel Rebollo Alcalá (Madrid).
- Manuel Palma Mariño (Cádiz).
- Juan Antonio Estop Zafra (Alcorcón-Madrid).
- Juan E. Jares Barredo (Oviedo-Asturias).
- José Luis Cañadilla López de Coca (Valls-Tarragona).
- Daniel Gastón di Angelo (Alfaz del Pi-Alicante).

TU PUEDES FORMAR PARTE DE

ERBE

¡Como lo oyes! Tú puedes estar informado antes que nadie de todas las novedades que aparecen en el mercado, tener acceso a programas exclusivos y a precios especiales, poder conseguir los trucos que te ayuden a salir de esa pantalla que se te resiste, regalos, pósters, sorteos, pegatinas... y un montón más de ventajas.

APUNTATE YA AL CLUB

No lo pienses más. Si te apuntas ahora al Club ERBE vas a recibir, además, como regalo de inscripción:



- El juego que más te guste de nuestro catálogo*



- Una fabulosa camiseta con el anagrama del Club.



- Tu carnet personal de socio, que te dará derecho al 10% de descuento en todos los juegos de ERBE que compres, durante un año, en cualquiera de los establecimientos que aquí al lado se relacionan (o a través del Club si en tu zona no existiera ninguno).



TODO POR SOLO 3.000 PTS.... UN AUTENTICO CHOLLO

Elige ya tu juego, corre al teléfono y llámanos al (91) 447 34 10 o escríbenos a ERBE SOTWARE C/. Santa Engracia, 17 · 28010 Madrid para hacerte socio o pedir más información.

* QUEDAN EXCLUIDOS LOS PROGRAMAS EN DISQUETE

MICRO Manía
Sólo para adictos

EL ZORRO

Esta semana la vamos a dedicar por entero a este juego de U.S. Gold, el cual, a la vista de las cartas recibidas, parece haber tenido mucha aceptación. La carta elegida ha sido la de Ricardo Osés, de Zaragoza, aunque muchas han sido las que nos han contado las claves para llegar hasta el final del juego.

1. Subimos por la parra, de un salto pasamos al pozo y bajamos por él.
2. Hábilmente pasamos el lago subterráneo saltando de un balón flotante a otro hasta llegar a la siguiente pantalla.
3. Nos dirigimos al rincón inferior izquierdo y bajamos por el hueco que hay.
4. Cogemos la planta y salimos por donde entramos.
5. Subimos hasta donde está la bola, la tocamos y nos colocamos en el ascensor con ella. Cuando el ascensor llegue al suelo la bola irá hasta el de la derecha abriendo así la puerta que protegía la copa. Ya podemos dejar la planta y marcharnos por el pozo.
6. Al salir del pozo coger el pañuelo para puntos extra.
7. Nos dirigimos a la habitación del sofá. Cogemos la llave que nos abrirá la puerta del 2.° piso y ya podemos acceder a la botella (esto hay que hacerlo siempre que vengamos a coger algún objeto a esta pantalla).
8. Dar la botella a uno de los mejicanos que están en el bar. Saltando sobre él llegamos a la casa. Subimos al piso superior y cuando salga un guardia por la puerta de la izquierda peleamos con él hasta que caiga y quede colgado de la lámpara. Volvemos al suelo y bajamos por la escalera para coger la copa.
9. Volvemos a la habitación del sofá y cogemos el hierro de marcar.
10. Vamos a la herrería (se accede subiendo por la palmera y saltando a la izq., o bien desde el 2.°, y el 3.° piso de la pantalla del pozo hacia la dch.). Colocamos el hierro en la fragua. Saltamos sobre el fuelle hasta que veamos cómo sale fuego que calienta el hierro. Lo cogemos y marcamos el burro. Ya podemos coger la herradura.
11. Vamos a la habitación del sofá de nuevo y cogemos la trompeta. La hacemos sonar sobre el trampolín de la pantalla de la palmera. Ya podemos coger la bota.
12. Volvemos a la habitación del sofá y cogemos ahora la campana, que colocaremos en la parte superior derecha de la casa que hay en la pantalla de la tumba.
13. Volvemos a la habitación del sofá y repetimos el paso anterior, pues aparece otra campana.
14. Al colocar las dos campanas se abre un pasadizo bajo la tumba, antes de entrar volvemos a la habitación del sofá y cogemos la llave.
15. Recorremos las pantallas subterráneas recogiendo las bolsas de dinero hasta que veamos que en una pantalla aparecen un estribo y una bota. Subimos por esa escalera y aparecemos en la cárcel.
16. Como tenemos la llave, podemos liberar a los prisioneros, que, conforme van saliendo, forman una escalera para que pasemos a otra pantalla.
17. Sin más peligros llegaremos donde está prisionera Lupita, pero aún no hemos completado la aventura. Al subir junto a ella somos despedidos fuera de la prisión.
18. Volvemos, por última vez, a la pantalla del sofá y cogemos la rosa.
19. Podemos repetir el camino conocido (tumba-sótanos-cárcel-Lupita) o bajar por el pozo al lago e ir por la esquina superior dch., donde antes había una bola, y salir directamente al sótano de la cárcel.
20. Con la rosa ya podemos reunirnos con nuestra amada y completar así la aventura.



ASTERIX AND THE MAGIC CAULDRON • Arcade • Melbourne

ASTERIX Y OBELIX TRAS EL CALDERO MAGICO

¿Quién no ha disfrutado alguna vez con las simpáticas aventuras de esta peculiar pareja de galos? ¿A quién le es extraña la inteligencia del pequeño Asterix o la fuerza del bonachón de Obelix? Tan sólo unas cuantas personas en el planeta podrán contestar afirmativamente a estas preguntas. Por eso, sin más preámbulos ni presentaciones, comencemos a matar romanos.



Buena, excelente presentación de Asterix y Obelix en el mundo de la informática. En los cómics son geniales; en el cine maravillosos y aquí, a pesar de encontrarse en un medio completamente hostil y nuevo para ellos, han sabido estar a la altura de las circunstancias. La figura se lleva en la sangre, pero Melbourne House también tiene su parte de mérito.

Todo en el programa es correcto. Desde la elección del argumento hasta la confección de los escenarios, pasando por la representación de los personajes o el tipo de diseño. La ambientación es, pues, perfecta, y podemos decir que nos encontramos ante un programa que podría haberse extraído perfectamente de las páginas de cualquiera de los cómics protagonizado por tan particular dúo.

La historia transcurre del siguiente modo. Por motivos que ahora no vienen al caso y que serían largos de explicar, Obelix ha ido a propinarle una soberana patada al caldero donde Panoramix, el barbado duida de la aldea, cocina la pócima mágica. Esta ha saltado en ocho pedazos que han quedado esparcidos por toda la comarca. Ya conoceréis la importancia que tiene dicho brebaje para los miembros de la aldea, por lo que Asterix y Obelix han tenido que ponerse en marcha rápidamente y salir en busca de los trozos antes de que se produzca una inevitable invasión romana.



Aquí comienza el juego. Nos encontramos en la aldea y salimos al bosque a la caza del caldero. Nuestros objetivos más inmediatos serán principalmente dos: comer y no ser derrotados por los romanos. De esta forma tendremos que preocuparnos, además de orientarnos en los bosques, campamentos romanos o la propia ciudad de Roma, de ir capturando jabalíes para convertirlos en suculentos asados y de estar rápidos de reflejos para enfrentarnos a los legionarios.

Cada vez que nos encontremos con uno de estos animales o cada vez que queramos pelear contra un soldado, la pantalla sufrirá una notable transformación. Aparecerá un gran recuadro en el que apareceremos, junto con nuestro oponente, ampliados a un gran tamaño. Este será el escenario de los combates y como si en un programa de lucha se tratase, tendremos que tratar de derrotar a nuestro adversario haciendo uso de los diferentes tipos de golpes que podemos llevar a cabo.

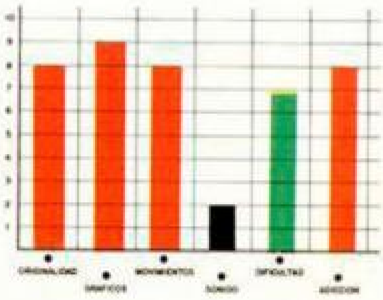
En Asterix y el Caldero

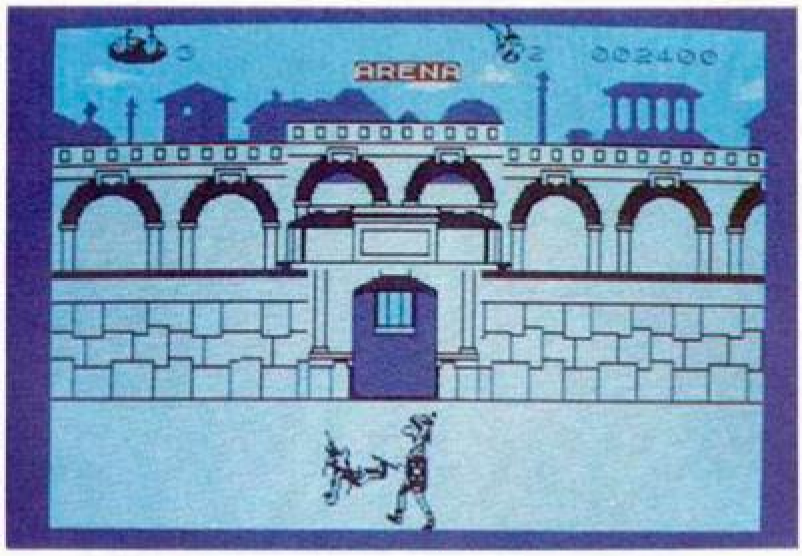
mágico se mezclan aspectos de arcade y videoaventura, pues también tendremos que ir recogiendo algunos objetos (llaves), que nos permitirán abrir nuevos caminos, aunque la mayor parte del juego consiste básicamente en un arcade en el que tenemos que ir derrotando a nuestros enemigos y tratando de conseguir el mayor número de puntos.

Si bien el juego puede resultar un tanto lento en su ejecución (tarda demasiado en realizar las pantallas y algunas veces los movimientos de Asterix disminuyen considerablemente en su velocidad), no ocurre lo mismo en su desarrollo, puesto que deberemos ser muy rápidos en nuestros movimientos, tanto en los combates como en las andanzas por bosques y campamentos.

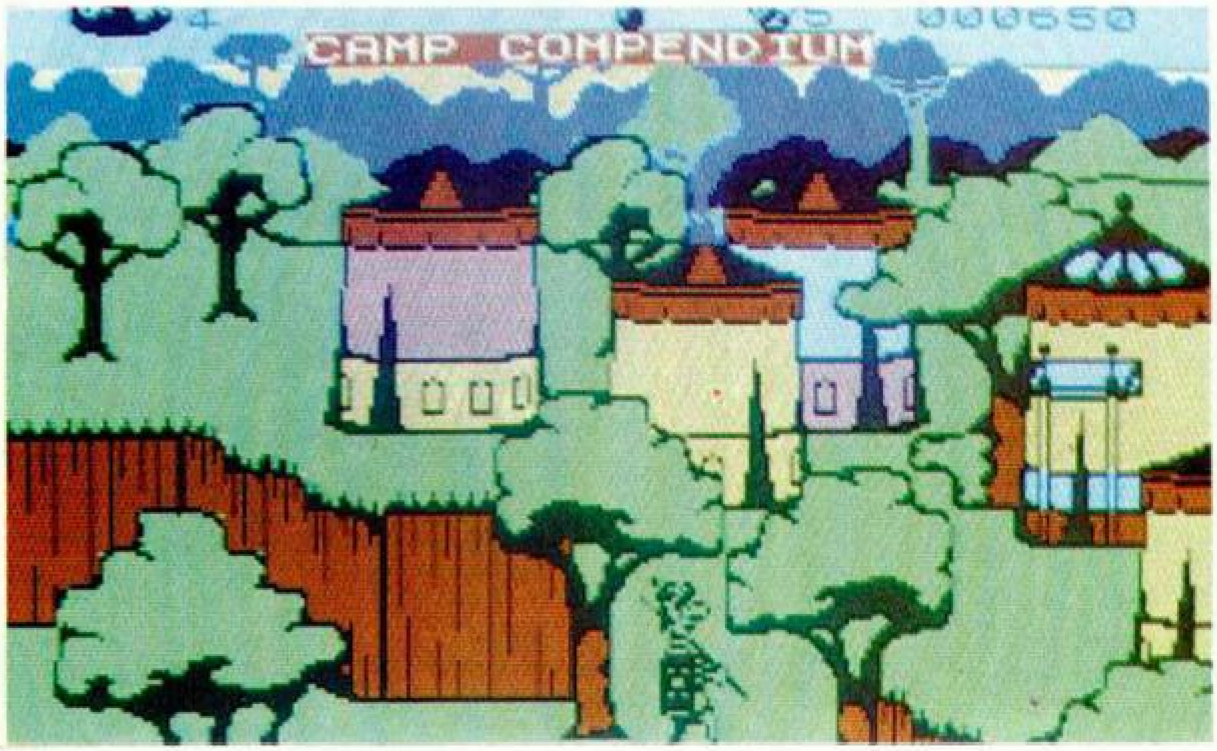
Uno de los aspectos más destacados del programa es, sin duda, el dedicado al diseño de las pantallas. El nivel de calidad alcanzado en cada una de ellas ha sido más que notable, y si antes decíamos que por el argumento parecía que estábamos metidos en un cómic, por los diseños aún mucho más, pues éstos son casi idénticos a los impresos sobre el papel.

Un gran programa que hace honor en todos sus aspectos a su prestigiosos protagonistas y que permitirá que Asterix y Obelix continúen haciendo las delicias de chicos y grandes.











Para empezar es muy importante que rebusquéis en cada pantalla, pues detrás de los árboles o las casas se encuentran escondidos los objetos, tales como llaves, comida o porciones del caldero.

Antes de introduciros en los campamentos romanos o en la propia Roma, aprovisionaros bien de jabalíes, pues en estos lugares se echa bastante de menos la existencia de algo que llevarnos a la boca.

Existen cuatro tipos de enemigos: jabalíes tontos, jabalíes astutos, legionarios y centuriones. A los primeros se les elimina con bastante sencillez; a los segundos cuesta un poco más, pues tendremos que recurrir en más de una ocasión al uso de nuestras potentes piernas. Un truco para acabar con los legionarios es colocarnos detrás de ellos y comenzar a darles patadas. Los centuriones ya son un caso aparte y, por lo general, será conveniente que los esquivemos y eludamos los combates frente a frente; suelen aparecer en las pantallas donde se encuentran dos soldados, y el acabar con ellos exige gran habilidad.

Otra opción es la de rendirnos. Esto se consigue simplemente colocándonos en un extremo del recuadro y permanecer completamente quietos. A los pocos segundos apareceremos en una de las mazmorras del circo romano, donde esperaremos para salir a la arena y enfrentarnos con un buen puñado de soldados.

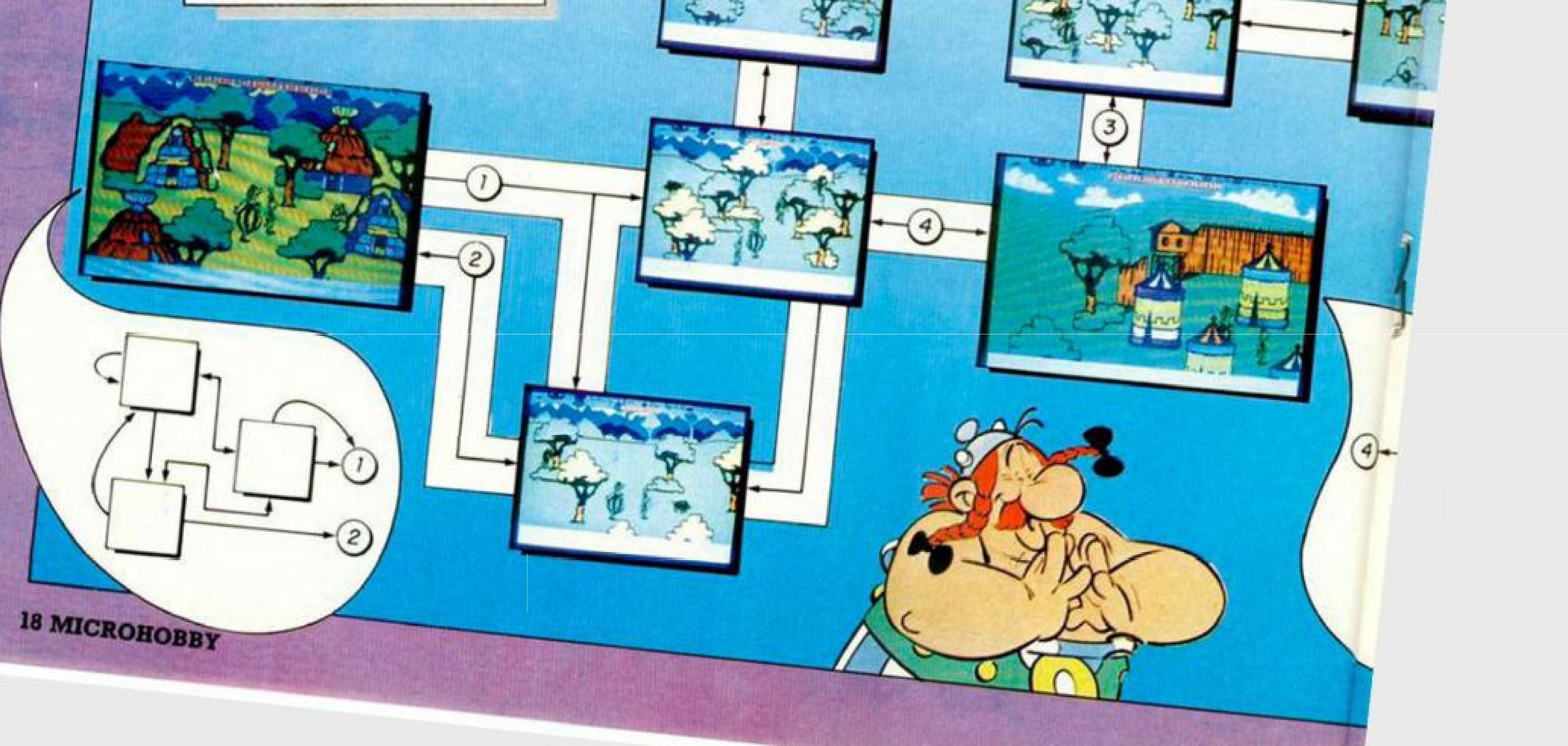
Esto de caer preso puede parecer arriesgado, pues una vez que entramos en la arena es muy difícil salir con vida de ella. El número de romanos que os atacará será considerable. Si sois hábiles conseguiréis derrotarlos a todos, pero es conveniente que, al menos en los primeros enfrentamientos, hagáis uso de la pócima mágica para haceros inmunes. Si logramos salir de allí victoriosos obtendremos como recompensa una de las porciones del caldero, aunque también puede ser útil para atajar el camino hacia Roma.

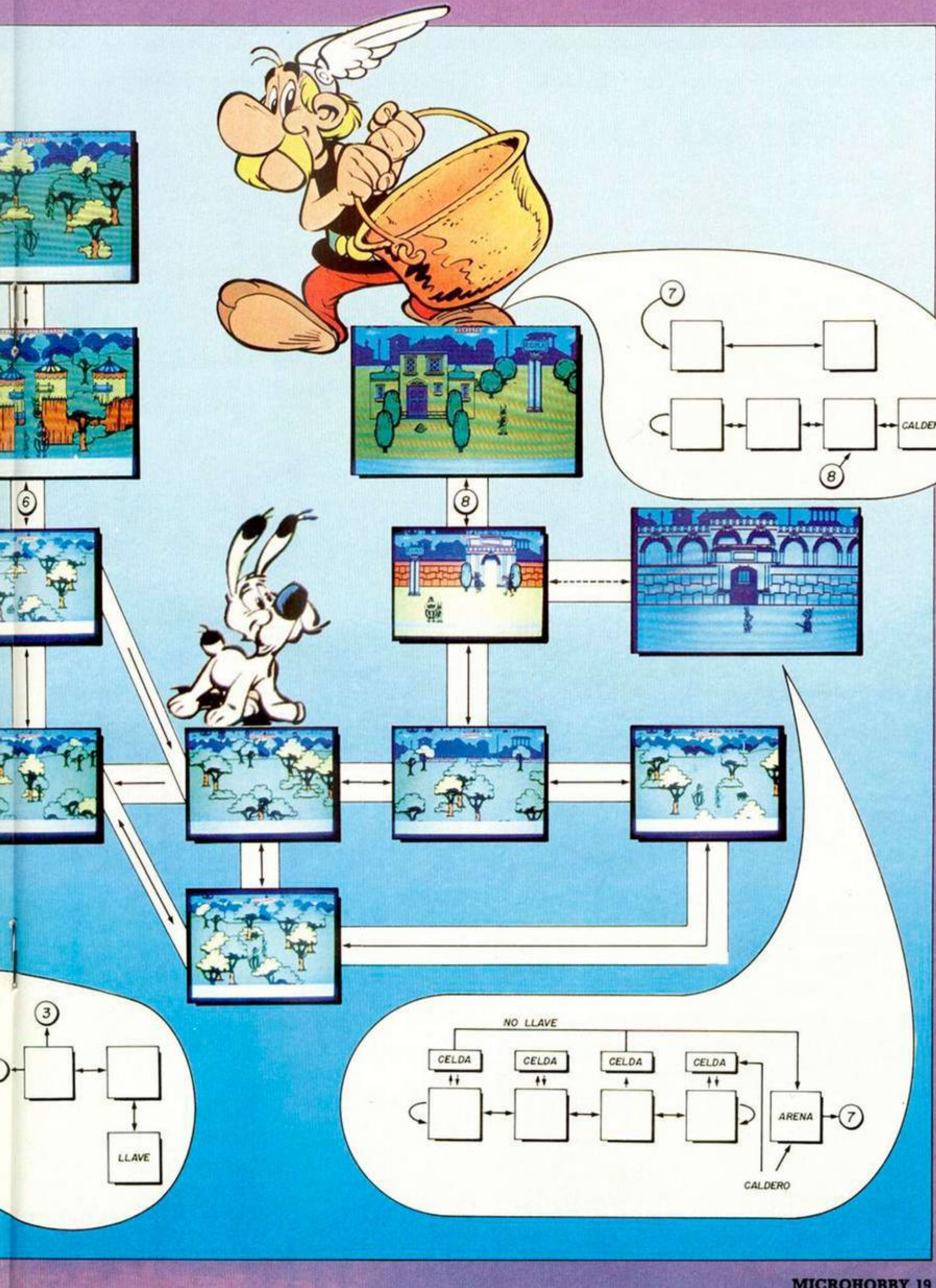
Astérix y el Caldero Mágico



A nuestros amigos Astérix y Obélix les ha sido encomendada una muy difícil misión. Se trata de recuperar el Caldero Mágico, que como todo el mundo sabe, es el principal instrumento utilizado por el druida de su aldea, Panorámix, para preparar la poción mágica. Sin ella, los galos no podrían mantener ni un solo día más su régimen de oposición a la dominación romana.

Con ayuda del mapa que te ofrecemos, podrás guiarles sin dificultad hasta el objetivo de su búsqueda.





7
CALDERO
8
6
8
3
LLAVE
NO LLAVE
CELDA
CELDA
CELDA
CELDA
ARENA
7
CALDERO

BOMSCARE • Videoaventura • Firebird

EL MIEDO A LA BOMBA



Bomscare pertenece a la colección Silver que Firebird ha comenzado a comercializar en las últimas fechas. En ella se incluyen un gran número de títulos, algunos originales y otros reposiciones, pero todos ellos de una mediana calidad. Sin embargo, su precio es francamente reducido, pues tenemos conocimientos de que en España costarán alrededor de 750 pesetas; unas ofertas interesantes debido a su excelente relación calidad-precio.

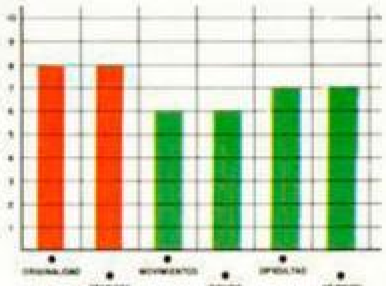
Un buen exponente de los títulos que se incluyen en este sello es Bomscare, el cual es, sin duda, uno de los juegos más destacados de la serie por su calidad y adicción.

Bomscare es una videoaventura tridimensional que se desarrolla en Neptuno. Un alienígena enemigo ha colocado una bomba de relojería en plena estación espacial y un robot, Arnold, ha sido programado para encontrar y desactivar dicho explosivo. Tú, por supuesto, controlas al robot.

El desarrollo del juego nos obligará a tener que ir recogiendo un buen número de objetos y utilizarlos convenientemente para conseguir ir acercándonos a nuestro objetivo. Todo ello, como cabría de esperar, esquivando y disparando contra los molestos alienígenas, no sólo ya para defender nuestra propia vida, sino también para obtener algo de diversión.

Lo más destacado del juego son sus excelentes diseños tridimensionales, los cuales resultan muy atractivos a la vista tanto por sus imaginativos decorados como por sus bellas formas.

Un juego bonito y adictivo que cuenta con el inconveniente inicial de que es difícil hacerse con el control de los movimientos del robot protagonista. A pesar de este detalle, un buen programa.



THE HAPPIEST DAYS OF YOUR LIFE • Videoaventura • Firebird

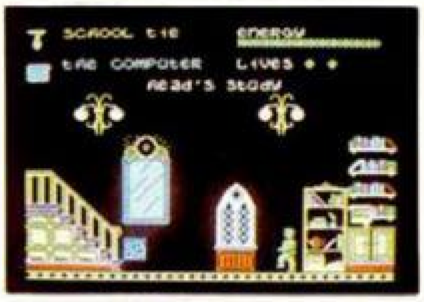
UNOS DIAS NO DEMASIADO FELICES

Este programa de largo título y mediano interés está en la línea de la videoaventura clásica. Sus escenarios y decorados en general nos recuerdan, quizás demasiado, a un programa que apareció hace casi un año con el título de That's the Spirit; pero desgraciadamente, al menos gráficamente, la calidad de The Happiest Days of your life, es ligeramente inferior.

Pero fijémonos en el programa que tenemos ante nosotros. «Deja tu reputación en limpio si no quieres recibir una auténtica paliza», reza el slogan del mismo. A partir de aquí se desata el argumento del juego. Según parece, la cartera del director del colegio ha desaparecido, y gracias a tu merecida fama de caco, todas las sospechas han recaído sobre tu gorda cabezota.

Ahora lo único que puedes hacer es recobrar esa dichosa cartera y demostrar al mundo entero tu inocencia.

Pero la cosa te va a costar. No te creas que vas a tener la carterita ahí, en la primera pantalla para que vayas tú y la cojas. No, te las vas a tener que ver y desear para encontrar la utilidad de cada uno de los cientos y cientos de objetos que te irás encontrando. Cada cosa debe ser llevada a su sitio y cada objeto debe estar en el lugar correspondiente. Así, si lo hacemos todo como Dios manda, podremos ir avanzando en el juego y llegar a encontrar la causa de nuestro mancillamiento.





En cuanto a la belleza plástica del asunto, os puedo decir, y vosotros podéis confirmar, que como mucho, un seis. No está mal de diseño, se ve que tiene trabajo, pero los resultados no son de una vistosidad excesiva.

Por último, en lo relativo a la diversión que nos puede ofrecer, debemos señalar que es un programa destinado a un tipo muy específico de gente, pues es necesario que el jugador esté dispuesto a probar una y otra vez hasta encontrar el uso de cada objeto. Se necesita, por tanto, invertir unas grandes dosis de paciencia mientras que a cambio se recibe una cantidad muy limitada de diversión.

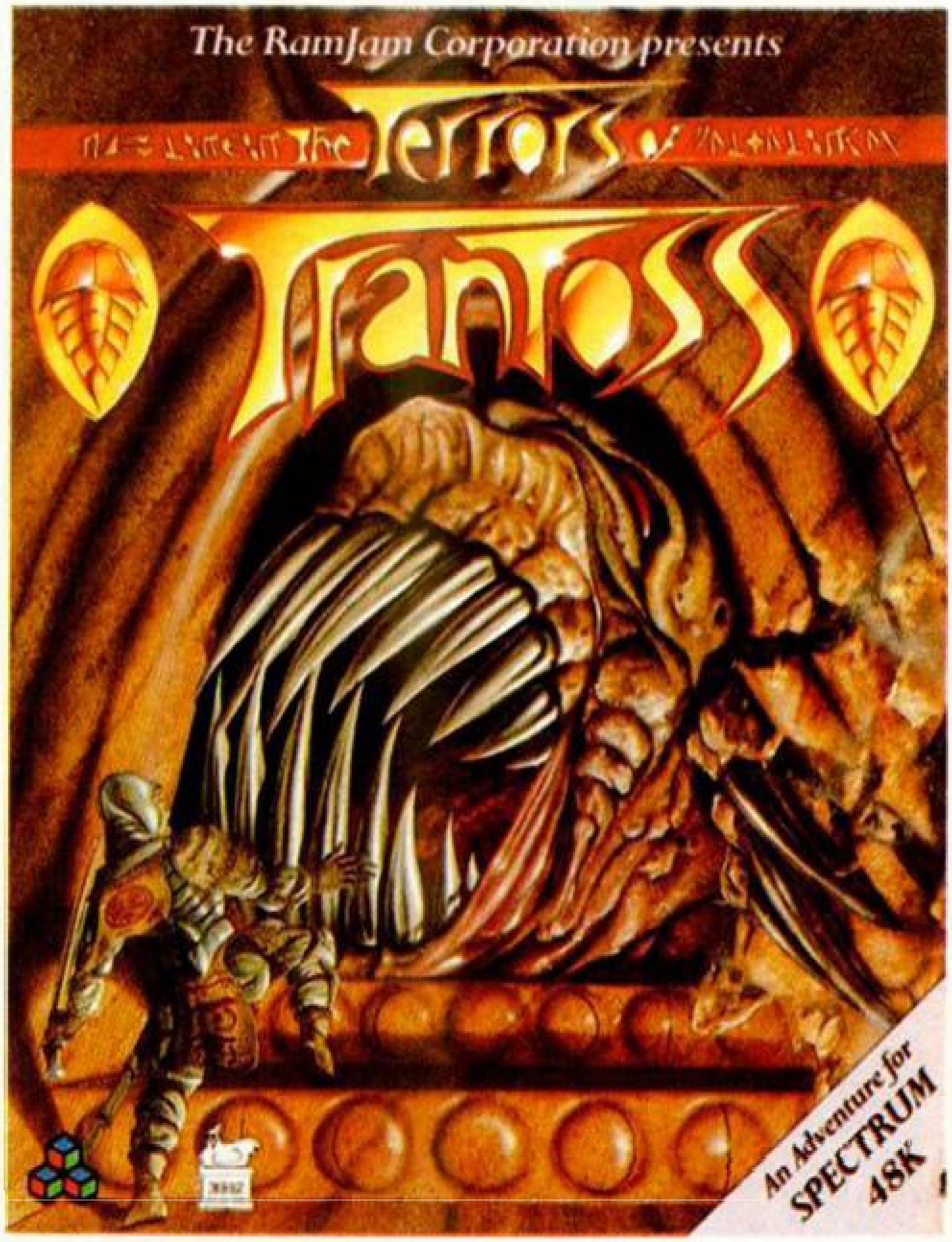


TERROR OF TRANTOS • Aventura • Ariolasoft

EL MISTERIO DE HAPSHAL

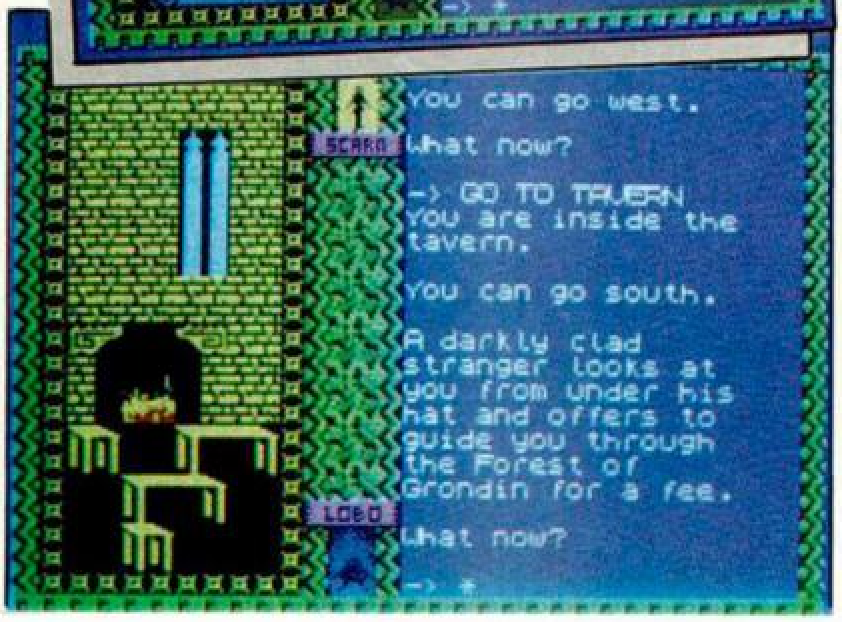
La aventura de texto es, como sabéis, un tipo de juegos completamente abandonado e ignorado en nuestro país. No vamos a entrar, una vez más, en las causas de este desprecio, pero desde luego nosotros debemos hacernos eco de toda la actualidad del software y, por supuesto, existen un número muy elevado de aventuras que poseen una contrastada calidad como para que no pasen desapercibidas.

Este es el caso de Terror of Trantos, un juego excelente. La acción se envuelve en un manto de misterio y nos transporta a la recóndita región de Hapshal, donde se encuentra un pequeño pueblo sobre el que se ha cernido una gran catástrofe: cada amanecer los









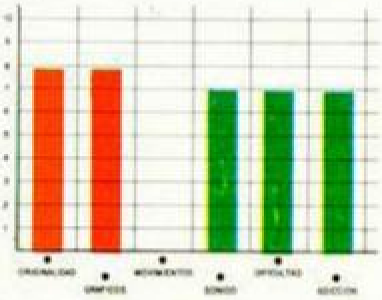
habitantes se dan cuenta de que uno o dos de sus conciudadanos ha desaparecido.

Dos hermanos, Lobo y Scarn, han sido convencidos para que tomen la responsabilidad de resolver el misterio. Emocionante y heroica misión que tú debes compartir.

El desarrollo del juego transcurre como cualquier otra aventura de texto: se establece un diálogo entre el jugador y el ordenador, en el transcurso del cual tendremos que sonsacarle la información necesaria para tomar las decisiones correctas que nos lleven a finalizar con éxito la aventura.

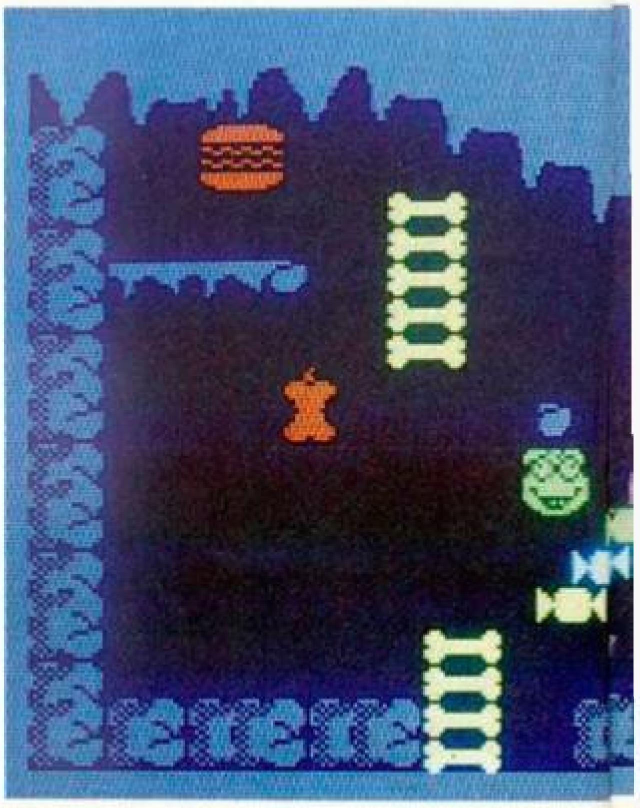
Terror of Trantos es una maravilla dentro de su género. Nada más verlo te das cuenta de que su calidad está muy por encima de la media. Los gráficos son, aunque desgraciadamente de un tamaño bastante reducido, de un diseño auténticamente artístico, y cualquiera de ellos podría competir por sí solo en el más prestigioso concurso de diseño gráfico por ordenador. Por otra parte, la historia se desarrolla de una manera muy fluida y divertida, resultando una verdadera delicia al ir leyendo (en inglés, of curse), los mensajes que aparecen ante nosotros y viendo estos magníficos dibujos.

Una de las mejores aventuras que hemos visto hasta el momento. Misteriosa, emocionante y adictiva.

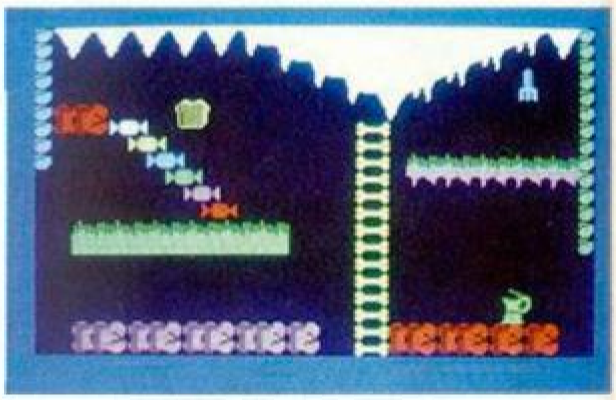


APRENDE A PROGRAMAR TU PROPIO JUEGO (IV)

Pablo ARIZA



En el capítulo de hoy vamos a desarrollar las rutinas que se encargarán de controlar el movimiento de todos los «bichos». Daremos vida a todos esos seres de pesadilla, que comenzarán a deambular por la pantalla.



Seguramente, habréis observado que en muchos juegos (sobre todo antiguos), al moverse los dibujos, éstos parpadean. La explicación es bien sencilla. Al mover un gráfico cualquiera, primero hay que borrar su antigua posición, y después dibujarlo en la nueva. Durante esta operación, puede haber un momento en que no quede en pantalla ni la antigua posición ni la nueva, con lo que parecerá que el gráfico ha desaparecido durante un momento, o que «parpadea». En principio, no parece haber forma de solucionar este problema. Sin embargo, la hay. Pero para entenderla, hay que hablar primero de la forma en que se genera la imagen en una televisión.

La pantalla se genera por una especie de cañón que dispara un haz de electrones. Este haz va pasando por toda la pantalla de izquierda a derecha y barriéndola por líneas horizontales. Cuando llega a la esquina inferior derecha, vuelve a la esquina superior izquierda para repetir el recorrido. Justo en ese instante se produce una interrupción enmascarable. Este proceso se repite 50 veces por segundo.

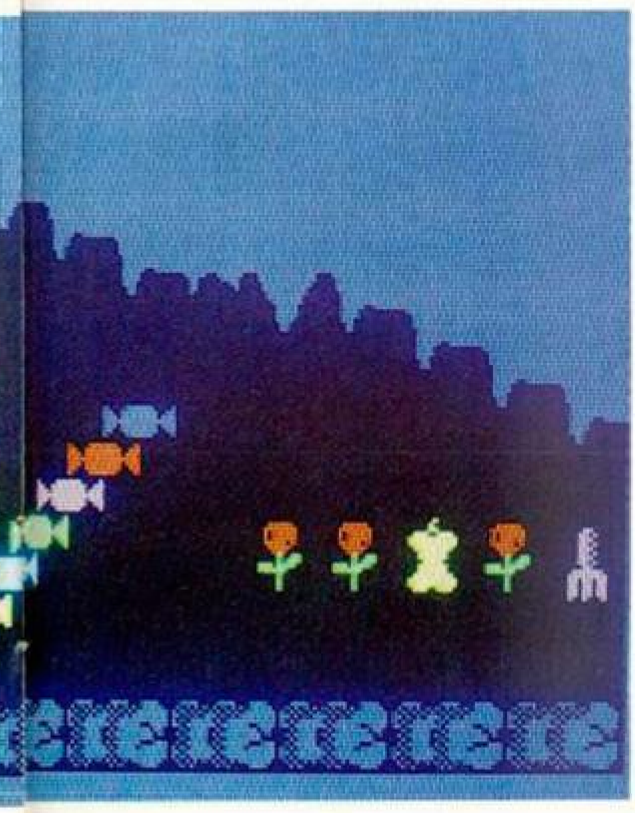
Ahora veamos qué relación tiene esto con el problema que abordamos. Si consiguiéramos que mientras que se borraran y dibujaran los gráficos, el haz de electrones no pasara en ningún momento por la zona de la pantalla donde se está efectuando la operación, no existiría ningún momento en el que no apareciera alguno de los gráficos: habríamos eliminado el parpadeo.



Naturalmente, no podemos obligar al haz de electrones a pasar por donde nosotros queramos, así que lo único que nos queda es esperar a que no esté en esa zona de pantalla y hacer entonces los dibujos. No podemos saber en cada momento dónde está el haz, pero sí sabemos cuándo está en la esquina superior izquierda, por producirse una interrupción. Este es un buen momento, pues ha de recorrer toda la parte superior del borde antes de llegar a la pantalla del ordenador propiamente dicha, que es donde tendremos que hacer los dibujos. Hay programas comerciales que utilizan rutinas de interrupción especiales para todo lo relacionado con el dibujo de los gráficos. Nosotros no haremos esto, ya que las interrupciones las usamos para la música. El método que usaremos nosotros es un tanto artesanal. La razón de escoger éste es por su sencillez, y no porque sea mejor, que no lo es.

Lo que hacemos es ejecutar una instrucción HALT antes de dibujar la nueva figura y borrar la antigua. La instrucción HALT espera a que se produzca una interrupción, con lo que, teóricamente, cuando se ejecute lo que viene después, el haz de electrones se encontrará aún en una zona del borde superior. Para utilizar este método, debemos hacer que no sea la rutina de control del movimiento de los «bichos» la que se encargue de dibujarlos, sino que ésta sólo se encargue de calcular las nuevas coordenadas del «bicho», la dirección en memoria del gráfico que corresponda, y el color, insertando después estos datos en un buffer. Cuando se tengan en el buffer los datos de todos los «bichos», se ejecutará el HALT y se dibujará todo lo que sea necesario. Recuérdese que al trabajar con OVER 1, para borrar una figura basta con dibujarla nuevamente, así que la subrutina no necesita diferenciar entre figuras a borrar y figuras a dibujar, sino que para ella todas serán iguales.

Hasta aquí todo parece ir bien, pero no todo es tan fácil. Cuando dijimos que la instrucción HALT esperaba una interrupción, no se mencionó que entonces el microprocesador pasa a ejecutar la rutina de interrupción, y no sigue con el programa hasta que no acabe. Esto quiere decir que si la interrupción dura demasiado, cuando estemos dibujando, el haz de electrones puede estar en la zona de la pantalla donde lo estemos haciendo, con lo que no habremos adelantado nada. Para colmo, nuestra ru-



tina de interrupción (la que se encarga de la música) puede tener 2 duraciones distintas, o sea, que el haz de electrones no estará siempre en el mismo sitio después de efectuar el HALT. Además, a no ser que logremos inventar una rutina que dibuje los gráficos sin consumir el tiempo, según vayamos haciendo dibujos (normalmente tendremos que hacer unos seis), el haz irá variando su posición en la pantalla.

Llegados a este punto tenemos dos soluciones para evitar el parpadeo. Una es renunciar a la música por interrupciones y hacer una rutina de ídem para dibujar las figuras. Otra es la solución artesanal. Nos decidimos por la última por estar dispuestos a mantener la música a toda costa. Dicha solución, no demasiado recomendable a pesar de ser la elegida, consiste en evitar el parpadeo de forma experimental. Si al definir una pantalla y probarla durante un cierto tiempo experimentalmente, tenemos la suerte de que no hay parpadeo, podemos tener la seguridad de que nunca va a haberlo. Ya sólo nos queda, por tanto, solucionar los casos en que no tengamos esa suerte y alguno de los «bichos» de la pantalla parpadeen. Lo que hacemos es sencillo: alteramos el orden en que se encuentran definidos en los datos de pantallas, con lo que serán dibujados en otro orden y el haz de electrones estará en otra posición al ser dibujados. Esto soluciona un 90 por 100 de los casos. En el resto podemos añadir un «bicho» más (siempre que no haya ya cuatro), y probar nuevos cambios en el orden. Si ni así lo conseguimos, tendremos que resignarnos o cambiar la colocación de los «bichos» hasta conseguir evitarlo. Como dato curioso, diremos que de las 36 pantallas, tan sólo una dio problemas serios para evitar el parpadeo. Se trata de la 33. Cuando esté terminado el juego, podréis ver cómo la dentadura existente en esa pantalla parpadea unas veces sí y otras no, según el momento en que se entre en la pantalla.

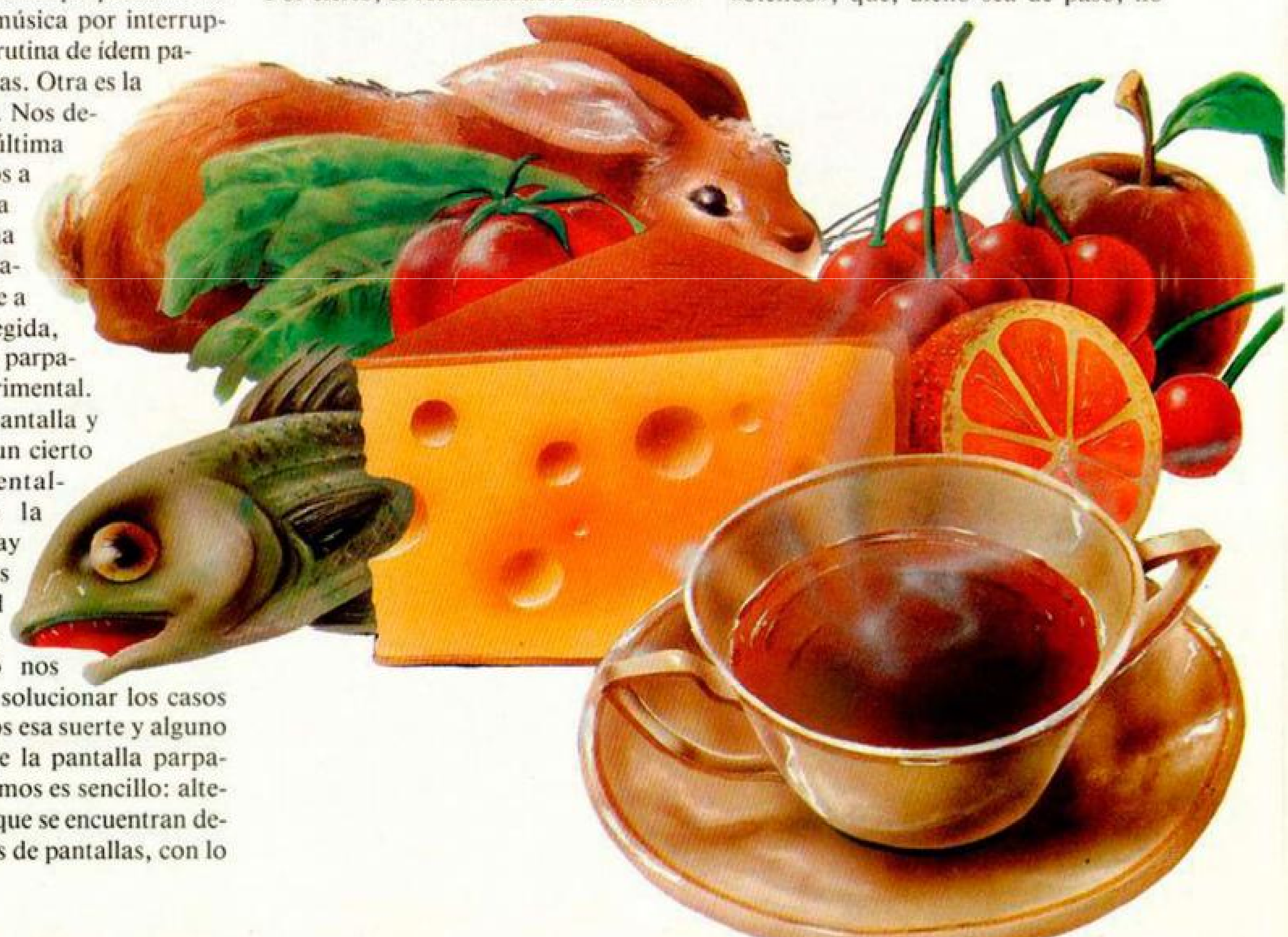
Los resultados obtenidos de todo esto, como podréis ver dentro de poco, son muy satisfactorios.

1	DD7E0E3CDDBE0FDA72F2	1421
2	DD4E00DD4601AFDD6E04	1101
3	DD6605CD7BF2DD7E0C3C	1317
4	DDBE0D3829DD7E02ED44	1175
5	DD7702DD7E03ED44DD77	1337
6	03DD6607DD6E09DD7409	1019
7	DD7507DD6608DD6E0ADD	1238
8	7508DD740AAFDD770CDD	1220
9	7E0BE607878747DD7E06	1068
10	3CB83801AFDD7706A7CB	1192
11	1FA7CB1F0F0F0FFE6020	859
12	023E204F0600DD6E07DD	740
13	660809DD7504DD7405DD	1024
14	7E02DD8600DD77004FDD	1123
15	7E03DD8601DD770147CD	1102
16	CBF2DD7E0B070707E607	1061
17	F640CD7BF2AFDD770E01	1410
18	1000DD09C9DDE5DD2AA0	1320
19	F2DD7502DD7403DD7100	1256
20	DD7001DD7704DD3605FF	1213
21	010500DD09DD22A0F2DD	1114
22	E1C9845B000000DD2184	1035
23	5BDD22A0F276DD4E0079	1286
24	3CC8DD4601DD6E02DD66	1208
25	03DD7E0408CDE7F20105	1046
26	00DD0918E3ED5BCA5B78	1222
27	923002ED44FE10D07993	1247
28	3002ED44FE10D03EFF32	1200
29	CF5BC978E6071E035320	1004
30	011D79E607200115E578	791
31	CD81223225F3780707E6	1110
32	03C658E56708424D772C	935
33	10FC69012000091DC208	646
34	F3D1E13E1008ED5335F3	1379
35	5E2356233E06A7280947	605
36	AFCB1BCB1A1F10F9444D	1075
37	21884F2C2CAE772D7AAE	970
38	772D7BAE77247CE60720	1009
39	0A7DC6206F38047CD608	882
40	675059EB083DC21BF3C9	1241

Movimiento de los «bichos»

En el listado en ensamblador que acompaña al artículo, se encuentran todas las rutinas necesarias para mover y dibujar los «bichos». Empecemos por la primera y más importante, etiquetada como SUMOVB. Al llamar a esta subrutina, el registro índice IX debe apuntar al comienzo de los datos del «bicho» en curso dentro de la tabla de trabajo. Por cierto, es recomendable tener delante la **figura 3** del capítulo pasado, a no ser que sepáis de memoria la función de todos los bytes de dicha tabla.

La subrutina SUMOVB se encarga del movimiento de un solo «bicho». Para moverlos todos hay que hacer un bucle en el que se vaya llamando a esta subrutina con IX señalando los datos en la tabla de trabajo de cada uno de los «bichos», que, dicho sea de paso, no



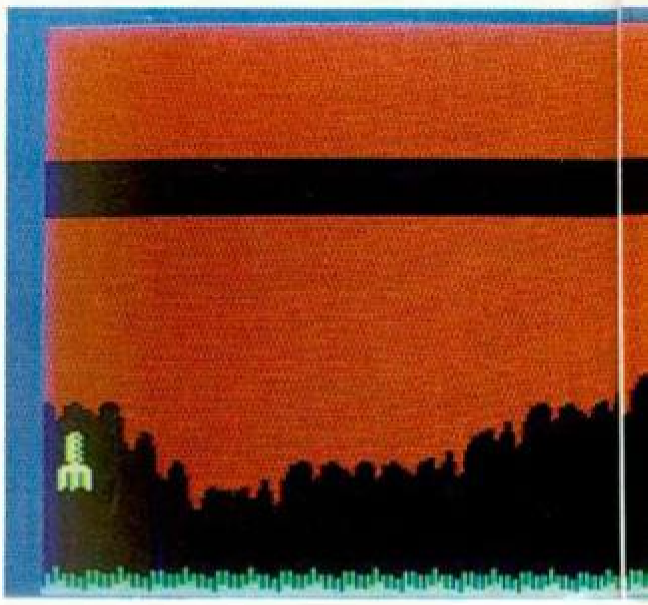
```
1   0000000003DB8734C4DB4 693
2   1E581A782DB473CE7FFE 1191
3   4FF26FF6700E3FFC1FF8 1398
4   07E0000000003DB8734C 667
5   4DB416681E782DB473CE 1079
6   7FFE4FF26FF6700E395C 1334
7   1C3807E0018003400380 642
8   03400380034003800340 463
9   018007E00FF00DB00DB0 993
10  0DB00DB0091000000000 403
11  1C703FF8575CAAAE9556 1209
12  AAAE555C4AAC555C4AAC 1190
13  555C4AAC55583FF00CF0 1151
14  3BFC778E77E6EFF3EFFB 1893
15  EFFBEFFB77FE77FE3BFC 2037
16  0CF00000000000000000 252
17  00000000000000000000 0
18  000F003F007800D801B0 591
19  01B0033003600660 0660 531
20  06601110199808880000 456
21  1998199819980007FFE 912
22  FFFFC003BFFD66661998 1530
23  7FFE7FFE088819981110 1116
24  00001998199819980000 531
25  7FFEFFFFC003BFFD6666 1734
26  19987FFE7FFE0CF005E0 1420
27  0200010007E00FF81CFC 777
28  1B7C3CFE3FEE3FD63EF6 1351
29  01EC1FFC0FF803E00CF0 1262
30  05E0020001 0007E00FF8 726
31  1CFC1B7C3CFE3FEE3FD6 1323
32  1EF601EC03FC0FF803E0 1258
33  0CF005E00200010007E0 715
34  0FF81CFC1B7C3CFE3FEE 1309
35  0FD600F601EC01FC03F8 1216
36  03E00F3007A000400080 649
37  07E01FF03F383ED87F3C 1086
38  77FC6BF06F0037803F80 1203
39  1FC007C00F3007A00040 716
40  008007E01FF03F383ED8 1027
41  7F3C77FC6BFC6F783780 1331
42  3FC01FF007C00F3007A0 955
43  0040008007E01FF03F38 813
44  3ED87F3C77FC6BFC6F7C 1430
45  37803FF81FF007C003C0 1159
46  0FF01FF81FF83C3C3BDC 1212
47  3FFC3FFC3FFC1DB81DB8 1371
48  1FF80FF00FF007E00180 1149
49  018007E00FF00FF01FF8 1149
50  1DB81DB83FFC3FFC3FFC 1371
51  3BDC3C3C1FF81FF80FF0 1212
52  03C003000780000000B40 408
53  2B505B685BA85BB45BB4 1119
54  5BDA5BDA0000FFFF0000 1128
55  7FFE3FFC018003C00000 1020
56  05A015A82DB42DB42DB4 1029
57  5BDA5BDA5BDA0000FFFF 1437
58  00007FFE3FFC00C001E0 1113
59  000002D00AD416DA15DA 911
60  2DDA2DDA5BDA5BDA0000 1144
61  FFFF00007FFE3FFCFFFC 1713
62  0002F7BDF7BDF7BD6319 1434
63  00036319F7BDF7BDF7BD 1435
64  0002FFFC000000000000 509
65  FFFC0002F7BDF7BDF7BD 1817
66  63190000300010019033D 217
67  67BDF7BDF782F03C07C0 1604
68  F8003FFF4000BDEFBDEF 1486
69  BDEF98C6C00098C6BDEF 1748
70  BDEFBDEF40003FFF0000 1238
71  000000003FFF4000BDEF 810
72  BDEFBDEF98C6C0000000 1526
73  98008CC0BDE6BDEF41EF 1683
74  3C0F03E0001F026001F0 672
75  0160010003800 7C007C0 627
76  0D600FE01FF03FF83BB8 1173
77  3C783FF81FF00EE00260 1098
78  01F00160010003800 7C0 669
79  07C00D600FE01FF03AB8 1060
80  38383D783FF81FF00EE0 1113
81  000000000000001FF870FC 643
82  3FCC3FCC1F8C1F9C1FF8 1171
83  10701F803FC03FC03FC0 1052
84  7FE00C00030000C00038 614
85  70FC3FCC3FCC1F8C1F9C 1256
86  1FF810701F803FC03FC0 1076
87  3FC07FE00000000000000 606
88  1FF83F0E33FC33FC31F8 1259
89  39F81FF80E0801F803FC 1110
90  03FC03FC07FE003000C0 1011
91  03001C003F0E33FC33FC 714
92  31F839F81FF80E0801F8 1152
93  03FC03FC03FC07FE0006 1032
94  0007000F001C00380080 282
95  03C007A01F407EA07F40 934
96  FE80FD40FA8055002800 1202
97  00020007000B003C0FF8 343
98  3FF87F307CD0FA90FAD0 1670
99  F5B0F2B0F760F8E07FC0 1973
100 3F0001C003600C981FFC 802
101 1B6C24923FFE5B6D2492 1016
102 7FFF5B6D24923FFE1B6C 1216
103 049003E0000000000000 375
104 00000000000000000630 54
105 077003601D5C0EB802A0 699
106 1D5C0F7801C000600080 673
107 168036180 0D86CC06B2C 895
108 016C360036D800580D80 662
109 0DB00030018001800600 501
110 01000168186C1B000336 322
111 34D63680006C1B6C1A00 717
112 01800D800C0001800180 636
113 800027E0581817E81008 782
114 1C081F081FC81FF81FF8 864
115 1FF80FF00FF00FF00FF0 1299
116 03C0000107E4181A17E8 736
117 1008103810F813F81FF8 906
118 1FF81FF80FF00FF00FF0 1323
119 0FF003C0000200050005 462
120 000D000A001A00360074 219
121 00EC01D803B807701CC0 979
122 73808E00F0008000C000 945
123 60000000300038003C00 260
124 3C001E001F000F8007C0 463
125 03F001F8007E000F000F 648
126 007E01F803F007C00F80 960
127 1F001E003C003C003800 237
128 300000006000C0008000 464
129 007001F403FA07D40FEA 1078
130 1F5530AA4E55512AAE94 942
131 AAA8AE9051604E403180 1152
132 0E000000000300380F78 253
133 1CE03BD037B837BE1B7E 1156
134 0CF80000FFFF00003FF8 1081
135 1FF0000007E01FF837FC 1088
136 6FE6FFFFEFF7FC3FFBDF 2126
137 73CE03C003C007E005A0 1107
138 06600FF00FF007E01FF8 1122
139 3FEC67F6FFFFEFF7FC3F 1959
140 FBDF73CE03C003C007E0 1416
141 05A006600FF00FF00000 777
142 1FF87FFEFFFFFFFF0000 1680
143 CCCC33330000CCCC3333 1020
144 0000FFFFFFFF7FFE1FF8 1680
145 1FF87FFEFFFFFFFF0000 1680
146 3333CCCC00003333CCCC 1020
147 0000FFFFFFFF7FFE1FF8 1680
148 0000055002A803F03DF8 807
149 3FF80E0009F83704577C 852
150 776E57B637CE08E60182 1128
151 0FA00F6002A802A803F0 869
152 3DF83FF80E0009F83704 950
153 577C776E576E379E08E6 1088
154 006203E003C0015402A8 775
155 03F03DF83FF80E0009F8 1134
156 3704577C76DE56DE373E 1035
157 08E6003206F806F00AA0 958
158 15400FC01FBC1FFC0070 906
159 1F9020EC3EEA76EE6DEA 1438
160 73EC6710418005F006F0 1154
161 154015400FC01FBC1FFC 879
162 00701F9020EC3EEA76EE 1207
163 76EA79EC6710460007C0 1097
164 03C02A8015400FC01FBC 876
165 1FFC00701F9020EC3EEA 1134
166 7B6E7B6A7CEC67104C00 1017
167 1F600F6009200548 07ED 600
168 1011482037C27FF49FF0 1156
169 7D707FE02E7823881D70 1066
170 11001E000E007B688D50 509
171 B7E017F07BE05C152FFE 1431
172 100011400 8061E740E3C 331
173 00400070000780088 0000 432
174 0000032807F4542A583A 566
175 BBF5A87DBC35BFBD483A 1476
176 586A2FF413480C3003C0 831
```

pueden pasar de cuatro en cada pantalla por razones de velocidad. Para empezar, incrementamos el contador de velocidad, y si no ha llegado al valor de ésta, salimos de la subrutina por NOND, pues el «bicho» aún no debe moverse. Si el contador llega al valor de la velocidad, procederemos a efectuar al movimiento. En primer lugar, introducimos en el **buffer** los datos de la figura que hay que borrar. Para ello se utiliza INSDIB, que veremos después, y a la que llamamos con las coordenadas en BC, el color en A, y la dirección de la figura en HL.

Ahora, ayudados por el contador de la longitud, comprobamos si ha llegado el momento de darse la vuelta. En caso negativo se salta a YEAH. En caso afirmativo cambiamos el signo de los incrementos haciendo NEG. Además intercambiamos la dirección de las figuras a derecha con la de las figuras a izquierda. De esta forma, el programa no tiene que preocuparse de la dirección en que se mueve el «bicho». Simplemente, coge lo que comenzó siendo la dirección de las figuras a izquierda, y que podrá ser de la izquierda o la derecha según el sentido que tenga el «bicho» en ese momento. De lo único que hay que tener cuidado es de que al entrar en una pantalla, los «bichos» direccionables comiencen andando hacia la izquierda (el incremento X debe comenzar siendo negativo), ya que de no ser así, andarían hacia atrás (esta circunstancia será después aprovechada en una pantalla).

Ahora vamos a YEAH, punto común tanto si se ha dado la vuelta como si no. Aquí se incrementa el contador de la longitud del recorrido (si acaba de darse la vuelta, se pone a cero). Después se cogen los tres primeros bits del duodécimo byte (recuérdese que dicho byte corresponde a IX + 11, y no a IX + 12). Estos bits corresponden al número de fases de animación. Ahora lo vamos a multiplicar por cuatro. La razón de esta sencilla multiplicación es para que el cambio de una fase de animación a otra sea cuatro veces más lento. Ahora vemos si hemos alcanzado la última fase, en cuyo caso regresaremos a la primera. Tras actualizar el contador dividimos su valor por cuatro. Esto es necesario por haber antes multiplicado por cuatro las fases de animación. El resultado lo multiplicamos por 32, ya que cada gráfico ocupa 32 bytes (16*16 = 256 pixels, 256/8 pixels por byte = 32 bytes). Si tras esta operación detenemos 96, es que se trata de la última fase de un «bicho» que tiene cuatro fases de animación. Pero en todos los «bichos» con cuatro fases, la cuarta es igual que la segunda, así que sustituimos el 96 por un 32. Para entender esto veamos un ejemplo:

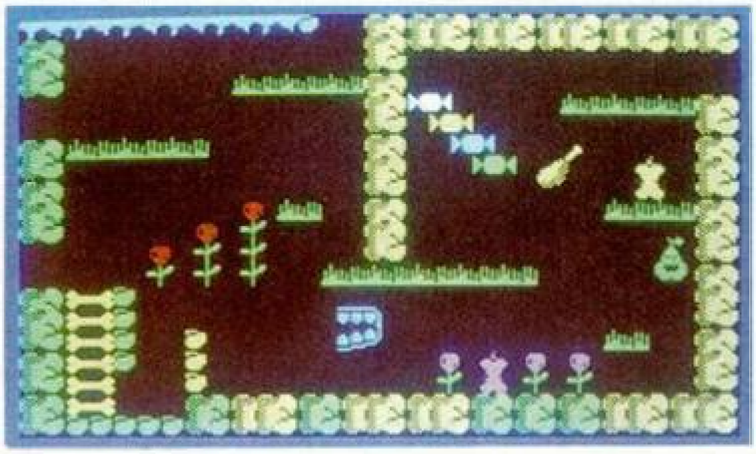
Supongamos que queremos hacer un muñeco que ande. Para ello, hacemos dos gráficos: Uno con la pierna izquier-



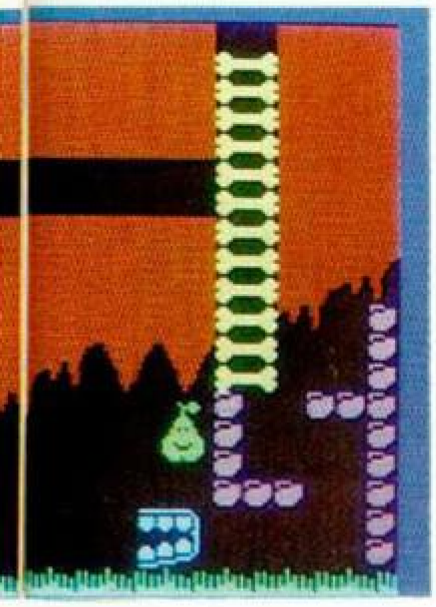


da delante y la derecha detrás, y otro con la pierna izquierda detrás y la derecha delante. Este tendría dos fases de animación. Pero si queremos que el movimiento sea más suave, añadimos una tercera fase con las dos piernas en el centro. Sin embargo, esta tercera fase habrá de ser intercalada dos veces, una entre las dos primeras fases y otra al final del ciclo para empalmar con el siguiente. Si llamásemos a la fase de las piernas juntas **fase 2**, y a las de una pierna delante y otra detrás, **fases 1** y **3**, el ciclo completo de animación sería 1, 2, 3, 2. Con esto vemos que al tener un «bicho» con **tres fases**, en realidad es de cuatro, y además, la cuarta es igual a la segunda.

Tras este pequeño paréntesis, continuamos por NONOV. El valor de la animación que habíamos multiplicado por 32, lo pasamos a BC. Ahora ponemos en HL la dirección de figuras a izquierda que, como vimos antes, no será siempre la de las figuras a izquierda. Al sumar HL y BC, calculamos la dirección del gráfico que corresponda teniendo en cuenta el estado de animación (al estar las figuras de las distintas fases unas a continuación de otras, sumando a la dirección de la primera un número igual a 32*(N—1), obtendremos la de la enésima figura). Esta dirección es la de la figura que vamos a dibujar, así que la próxima vez que movamos este «bicho», será la de la última figura dibujada. Por eso hay que guardarla en los bytes quinto y sexto de la tabla de trabajo (referidos a IX + 4 e IX + 5). Ahora actualizamos las coordenadas, sumándoles sus incrementos respectivos y colocándolas, además, en BC (coordenada X en C, coordenada Y en B). Después llama a COMCHO (cuyo cometido será el de comprobar si nuestro personaje ha colisionado con algún «bicho». Por último cargamos en A el color del gráfico, que estaba en los tres bits superiores del duodécimo byte, y lo hacemos brillante con OR 64. Ahora metemos estos datos en el **buffer** de la misma forma que hicimos con la figura antigua que había de ser borrada. Por último, pone a Ø el contador de velocidad y actualiza IX para que la próxima vez que se llame a SUMOVB, apunte a los datos del siguiente «bicho».







La pantalla 33 es la única que presenta problemas de parpadeo.

INSDIB, DIALL y COMCHO son rutinas tan simples que no vale la pena explicarlas con detalle. Tan sólo haremos unas apuntaciones.

INSDIB se encarga de introducir en el buffer los datos de una figura, para borrarla o dibujarla. El buffer comienza en 23428 y su puntero es BFPOS. Cada elemento tiene 5 bytes, 2 para las coordenadas, 2 para la dirección del gráfico y 1 para el color.

DIALL dibuja todas las figuras. Para ello, toma los datos del buffer y llama a DIMASU, de la que hablaremos enseguida.

Como ya dijimos antes, COMCHO se encarga de comprobar si *Jaime* se encuentra con algún «bicho». CORS es donde guardaremos las coordenadas de nuestro personaje cuando hagamos la rutina que lo controla. La variable CHOFLA será utilizada también por dicha rutina, y debe valer 255 si chocamos con algún «bicho».

Ya sólo nos queda ver DIMASU, la subrutina que se encargará de dibujar los gráficos en la pantalla.

Al entrar en DIMASU, HL contiene la dirección del gráfico, BC las coordenadas y el registro alternativo A', el color. En primer lugar debemos colocar en E y en D el número de filas y columnas respectivamente, sobre las que está el gráfico. Esto nos sirve para dibujar los atributos. Siempre que alguna de las dos coordenadas sea un múltiplo entero de ocho, el gráfico ocupará en esa coordenada dos posiciones de baja resolución. En caso contrario, ocupará tres. Hacemos la comprobación con ambas coordenadas y colocamos los valores correspondientes en DE. Según la posición del gráfico, tendremos que rellenar con atributos cuadra-

```
   10 PAPER 0: INK 7: BORDER 0: C
LEAR 32767: LOAD "c■4 3"CODE 611
83,4353: LOAD "c■4 4"CODE 44576,
7844
   20 FOR X=40000 TO 40060: READ
A: POKE X,A: NEXT X: POKE 23606,
32: POKE 23607,245: POKE 23658,8
   30 DATA 217,229,42,0,91,34,1,2
39,34,3,239,62,238,237,71,237,94
,42,118,92,205,92,243,118,175,21
9,254,246,224,60,32,28,221,33,0,
91,221,126,0,60,40,5,205,204,241
,24,245,205,165,242,24,227,225,2
17,62,63,237,86,237,71,201
   40 INPUT "PANTALLA? ";N,"MUSIC
A? ";M: RANDOMIZE M: POKE 23296,
PEEK 23670: POKE 23297,PEEK 2367
1: RANDOMIZE N: PRINT : RANDOMIZ
E USR 40000: GO TO 40
```

LISTADO ENSAMBLADOR

```
*C-
*D+
SUMOVB  LD   A,(IX+14)
        INC  A
        CP   (IX+15)
        JP   C,NOND
        LD   C,(IX+0)
        LD   B,(IX+1)
        XOR  A
        LD   L,(IX+4)
        LD   H,(IX+5)
        CALL INSDIB
        LD   A,(IX+12)
        INC  A
        CP   (IX+13)
        JR   C,YEAH
        LD   A,(IX+2)
        NEG
        LD   (IX+2),A
        LD   A,(IX+3)
        NEG
        LD   (IX+3),A
        LD   H,(IX+7)
        LD   L,(IX+9)
        LD   (IX+9),H
        LD   (IX+7),L
        LD   H,(IX+8)
        LD   L,(IX+10)
        LD   (IX+8),L
        LD   (IX+10),H
        XOR  A
YEAH    LD   (IX+12),A
        LD   A,(IX+11)
        AND  7
        ADD  A,A
        ADD  A,A
        LD   B,A
        LD   A,(IX+6)
        INC  A
        CP   B
        JR   C,WEAN
        XOR  A
WEAN    LD   (IX+6),A
        AND  A
        RR   A
        AND  A
        RR   A
        RRCA
        RRCA
        RRCA
        CP   96
        JR   NZ,NONOV
        LD   A,32
NONOV   LD   C,A
        LD   B,0
        LD   L,(IX+7)
        LD   H,(IX+8)
        ADD  HL,BC
        LD   (IX+4),L
        LD   (IX+5),H
        LD   A,(IX+2)
        ADD  A,(IX+0)
        LD   (IX+0),A
        LD   C,A
        LD   A,(IX+3)
        ADD  A,(IX+1)
        LD   (IX+1),A
        LD   B,A
        CALL COMCHO
        LD   A,(IX+11)
        RLCA
        RLCA
        RLCA
        AND  7
        OR   64
        CALL INSDIB
        XOR  A
NOND    LD   (IX+14),A
        LD   BC,16
        ADD  IX,BC
        RET
;
;
INSDIB  PUSH IX
        LD   IX,(BFPOS)
        LD   (IX+2),L
        LD   (IX+3),H
        LD   (IX+0),C
        LD   (IX+1),B
        LD   (IX+4),A
        LD   (IX+5),255
        LD   BC,5
        ADD  IX,BC
        LD   (BFPOS),IX
        POP  IX
        RET
;
BFPOS   DEFW 23428
;
        ORG  62117
;
DIALL   LD   IX,23428
        LD   (BFPOS),IX
        HALT
VEL     LD   C,(IX+0)
        LD   A,C
        INC  A
        RET  Z
        LD   B,(IX+1)
        LD   L,(IX+2)
        LD   H,(IX+3)
        LD   A,(IX+4)
        EX   AF,AF'
        CALL DIMASU
        LD   BC,5
        ADD  IX,BC
        JR   VEL
;
;
COMCHO  LD   DE,(CORS)
        LD   A,B
        SUB  D
        JR   NC,POY
        NEG
POY     CP   16
        RET  NC
        LD   A,C
        SUB  E
        JR   NC,POX
        NEG
POX     CP   16
        RET  NC
        LD   A,255
        LD   (CHOFLA),A
        RET
;
;
DIMASU  LD   A,B
        AND  7
        LD   E,3
        LD   D,E
        JR   NZ,EITR
        DEC  E
EITR    LD   A,C
        AND  7
        JR   NZ,DITR
        DEC  D
DITR    PUSH HL
        LD   A,B
        CALL 8881
        LD   (RRRR+1),A
        LD   A,B
        RLCA
        RLCA
        AND  3
        ADD  A,#58
        PUSH HL
        LD   H,A
        EX   AF,AF'
PLOE    LD   B,D
        LD   C,L
PLOC    LD   (HL),A
        INC  L
        DJNZ PLOC
        LD   L,C
        LD   BC,32
        ADD  HL,BC
        DEC  E
        JP   NZ,PLOE
        POP  DE
        POP  HL
        LD   A,16
CLAP    EX   AF,AF'
        LD   (RECOUP+1),DE
        LD   E,(HL)
        INC  HL
        LD   D,(HL)
        INC  HL
RRRR    LD   A,0
        AND  A
        JR   Z,NOROT
        LD   B,A
        XOR  A
BLOB    RR   E
        RR   D
        RRA
        DJNZ BLOB
NOROT   LD   B,H
        LD   C,L
RECOUP  LD   HL,0
        INC  L
        INC  L
        XOR  (HL)
        LD   (HL),A
        DEC  L
        LD   A,D
        XOR  (HL)
        LD   (HL),A
        DEC  L
        LD   A,E
        XOR  (HL)
        LD   (HL),A
        INC  H
        LD   A,H
        AND  7
        JR   NZ,RPRDO
        LD   A,L
        ADD  A,32
        LD   L,A
        JR   C,RPRDO
        LD   A,H
        SUB  8
        LD   H,A
RPRDO   LD   D,B
        LD   E,C
        EX   DE,HL
        EX   AF,AF'
        DEC  A
        JP   NZ,CLAP
        RET
CORS    EQU  23498
CHOFLA  EQU  23503
```

dos de 2X2 ó 3X3, o rectángulos de 2X3 ó 3X2. Cuando sabemos las dimensiones del cuadro a rellenar de atributos, calculamos la dirección de ídem donde este cuadro comienza. Hay una rutina de la ROM en la dirección 8874 (22AAh), que calcula la dirección de pantalla de unas determinadas coordenadas en alta resolución. Nosotros la llamamos por la dirección 8881 porque nuestras coordenadas tienen origen en la parte superior de la pantalla, en lugar de la inferior. La entrada por ese punto requiere que esté copiada en A la coordenada Y. Al salir de la rutina, A tiene el número de bit dentro del byte que corresponde al pixel deseado, pero en nomenclatura inversa: Ø para el bit más significativo y 7 para el menos significativo. Este valor lo guardamos en RRRR + 1 para su posterior utilización. Probablemente os suene eso de sumarle 1 a una etiqueta. Si no es así, repasaos los anteriores capítulos. Por otra parte, tenemos en HL la dirección de pantalla donde se encuentra el pixel. Da la casualidad de que el byte bajo de la dirección de pantalla y el de la de atributos, son iguales, así que ahora sólo hemos de calcular el byte alto de esta úl-

tima. Este byte debe ser 88 + INT (coordenada Y/64). Una vez calculado, guardamos la dirección de pantalla en la pila y la convertimos en la dirección de atributos. Ahora pasamos a A el color, que estaba en A', y procedemos al rellenado de color. Sobre PLOE se cierra el bucle para cada fila, y sobre PLOC se cierra el de cada columna que ocupa dicha fila. Tras terminar con los atributos entramos en el dibujo propiamente dicho. Recuperamos de la pila la dirección de pantalla en DE, y la dirección en memoria del gráfico en HL. Cargamos A con 16 y entramos en el bucle para cada uno de los 16 pixels que tiene el gráfico de alto. Guardamos el contador en A' para poder utilizar A libremente. Guardamos DE para su posterior uso, porque ahora lo necesitamos. Cargamos en DE los dos bytes que componen los 16 pixels horizontales que tiene el gráfico de ancho y actualizamos HL para que la próxima vez leamos los bytes de la siguiente línea de pixels. En RRRR se carga en A el número de rotaciones que hay que efectuar a los bytes contenidos en DE para que la esquina izquierda del gráfico coincida con el pixel indicado por la coordenada X. Este valor había sido colocado antes en RRRR + 1. Si no hay que hacer ninguna rotación, nos saltamos la parte correspondiente del listado. Si hay que hacerlas se utiliza A para recoger los bits que se salen de D. Ahora guardamos en BC la dirección de la figura para recuperar en HL la dirección de pantalla. Metemos los contenidos de E, D y A en la dirección indicada por HL y consecutivas, utilizando XOR para el OVER 1. Actualizamos la dirección de pantalla para que apunte a la siguiente línea de pixels y, por último, hacemos que de nuevo HL tenga la dirección del gráfico y DE la de pantalla, cerrando después el bucle.

El cargador

Con los listados de hoy, podremos ver los resultados prácticos de tanta teoría. Para poder probarlo hacer lo siguiente:

— Teclear con el cargador universal de código máquina los listados **1** y **2** haciendo Dump en 40000. La longitud del primero es de 400 bytes y la del segundo, 1760. Grabarlos como «cm4 1» y «cm4 2».

— Teclear el **programa 1** y grabarlo en cinta con LINE 10.

— Borrar la memoria y cargar el programa del segundo capítulo. Cuando finalice la carga, pararlo con STOP y cargar el programa del tercer capítulo (sin borrar la memoria). Cuando termine de cargarse pararlo también con STOP y cargar los dos bloques tecleados hoy con:

```
LOAD«cm4 1»CODE 61900,400:
LOAD «cm4 2»CODE 63776,1760
```

— Grabar todo a continuación del BASIC tecleado hoy con:

```
SAVE«cm4 3»CODE 61183,4353:
SAVE«cm4 4»CODE 44576,7844
```

— Borrar la memoria y cargarlo todo desde el principio con LOAD"".

Una vez finalizada la carga, se nos pedirán dos números, uno de pantalla y otro de música. Los números a introducir han sido publicados en tablas en capítulos anteriores. Una vez introducidos, y si nos hemos equivocado, veremos la pantalla deseada con los «bichos» moviéndose mientras suena la música que hayamos elegido. Que lo disfrutéis.

EL SISTEMA "FILMATION" (y IV)

José Manuel LAZO

Terminamos con este capítulo esta interesante serie sobre el sistema "Filmation" y, particularmente, sobre la gestión de sprites en alta resolución, lo que os permitirá crear en vuestros propios programas movimientos pixel a pixel, además de incorporar la sensación de tridimensión, ya que los sprites pasan por encima del fondo sin alterarlo.



Tal y como pudimos apreciar en el pasado capítulo, las rutinas de movimiento a utilizar son distintas según la dirección del desplazamiento. Las más sencillas y rápidas son las de movimiento horizontal, es decir, cuando los personajes no tienen que desplazarse ni hacia arriba ni hacia abajo. Por esto, cuando no utilicemos movimientos verticales puede agilizarse bastante todo el proceso ganando en rapidez.

Sin movimiento vertical

En el caso de que el muñeco no se mueva en vertical hay que hacer una serie de operaciones mucho más sencillas: basta con intercambiar el sprite antiguo por el moderno con un solo bucle.

Llegados a este punto hay que destacar varias características de nuestra rutina de sprites:

— Deberemos utilizar tantos buffers como sprites vayamos a mover por la pantalla, es decir: podemos tener cien gráficos en memoria, pero si sólo vamos a usar uno cada vez, es necesario tener sólo un buffer para el mismo, así como el complementario *(caso de que vayamos a usar la rutina MOVE).*

— La rutina MOVE la podemos usar para cualquier desplazamiento horizontal por grande que sea, sin embargo, no se puede utilizar para un desplazamiento vertical mayor que el número de scanes que tenga el sprite. Si tuviéramos que moverlo a tanta distancia tendríamos que borrarlo previamente con la rutina SPRITB y luego imprimirlo en la nueva posición con la rutina SPRITE.

— MOVE sólo funcionará coherentemente en el caso de que la tabla de cada gráfico esté inicializada por SPRITE. SPRITB tiene las mismas características, es decir: la primera rutina que se ha de usar siempre para un gráfico determinado es SPRITE.

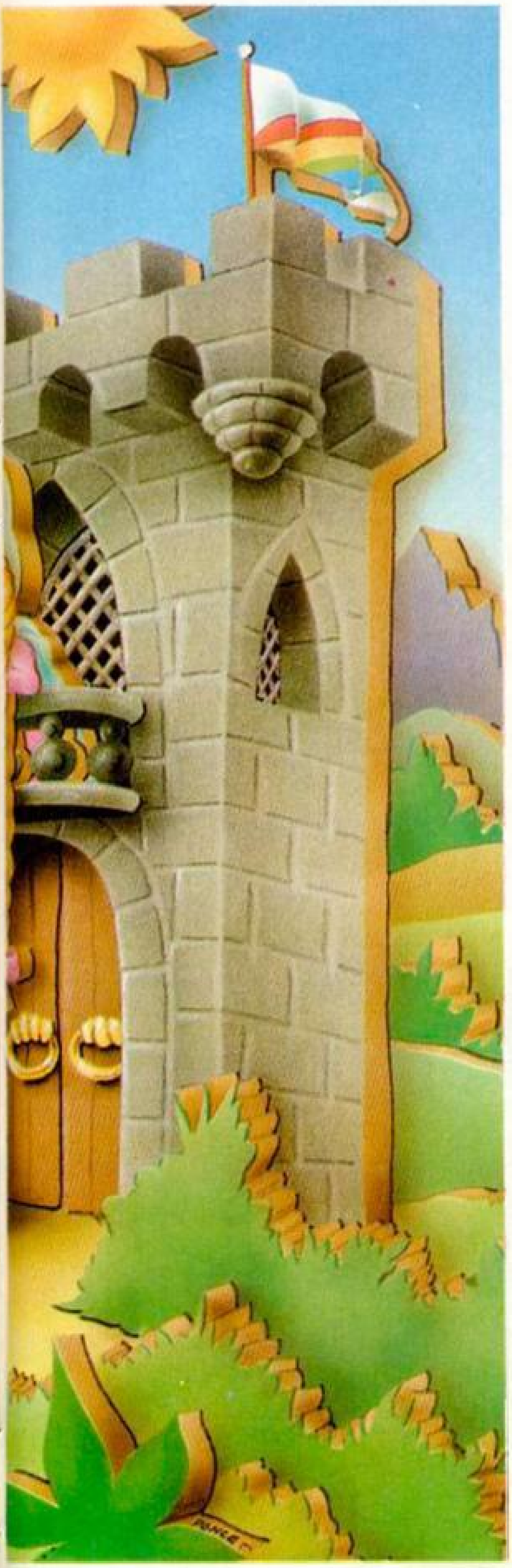
— No se ha puesto ningún tipo de chequeos para evitar una ralentización innecesaria en la mayoría de los casos; esto quiere decir que si nos salimos de la pantalla por la parte superior o inferior en cualquier aplicación que le demos a nuestras rutinas de sprites ésta se colgará irremisiblemente. Si nos salimos por los laterales no se produce un efecto tan trágico, sin embargo, no funcionará bien.

— Todas las rutinas aquí expuestas son desarrollos totalmente originales y los algoritmos usados en la misma son de nueva concepción, pudiendo utilizarse los mismos con fines didácticos o de programación para nuestra Revista.

Ejemplo de utilización

En el **listado 4** se encuentra un pequeño programa que se encarga de gestionar estas rutinas en un ejemplo de utilización. Sólo es necesario teclear este programa a continuación de las rutinas en el archivo fuente y cargar en la memoria del ordenador los dos gráficos que vayamos a utilizar, en este caso el coche de la **figura 4.**

Las datas situadas en el **listado 5** son las necesarias para crear este coche. Sólo es preciso, con la ayuda del cargador universal de CM. teclearlas y, luego, realizar un DUMP en la dirección 60000. Cuando lo tengamos, salvamos los bytes en una cinta con la orden SAVE del cargador especificando como comienzo la 60000, como longitud 1024



octetos y como nombre uno cualquiera.

Cuando tengamos estos bytes, ensamblamos el listado fuente compuesto por la rutina de sprites y el ejemplo de utilización y volvemos al Basic para cargar los gráficos del coche en la dirección 63ØØØ, especificada en las tablas de gráficos.

Por supuesto, si no os gusta esta figura podéis cambiarla por cualquier otra, siempre que se respete el tamaño.

Con un RANDOMIZE USR 6Ø749 podremos ejecutar el ejemplo. Las teclas de movimiento del coche son O, P, Q, A, y Space, al igual que en la primera «Demo».

Ahora podéis apreciar la suavidad del movimiento, pero si antes de ejecutar el programa con el RANDOMIZE apropiado, hacéis un bucle de Basic tal como **FOR n = 1 TO 7Ø4:PRINT "A";:NEXT n** podréis ver también el efecto **Sprite**, que permite desplazar la figura por la pantalla sin que se estropee el fondo.

Podéis enviar vuestras sugerencias para futuras ampliaciones sin dudar de que serán bien acogidas.

LISTADO 4

DESENSAMBLE DE LA RUTINA

```
439Ø GRAFI1 DEFW 63ØØØ
44ØØ        DEFW Ø
441Ø        DEFW Ø
442Ø        DEFW Ø
443Ø        DEFB Ø
444Ø        DEFB Ø
445Ø        DEFW 63256
446Ø        DEFW Ø
447Ø        DEFB 8
448Ø        DEFB 32
449Ø        DEFW 64Ø24
45ØØ        DEFB Ø
451Ø        DEFW 64352
452Ø        DEFW Ø
453Ø        DEFW Ø
454Ø        DEFW Ø
455Ø        DEFW Ø
456Ø        DEFB Ø
457Ø        DEFB Ø
458Ø GRAFI2 DEFW 63512
459Ø        DEFW Ø
46ØØ        DEFW Ø
461Ø        DEFW Ø
462Ø        DEFB Ø
463Ø        DEFB Ø
464Ø        DEFW 63768
465Ø        DEFW Ø
466Ø        DEFB 8
467Ø        DEFB 32
468Ø        DEFW 64Ø24
469Ø        DEFB Ø
47ØØ        DEFW 64352
471Ø        DEFW Ø
472Ø        DEFW Ø
473Ø        DEFW Ø
474Ø        DEFW Ø
475Ø        DEFB Ø
476Ø        DEFB Ø
477Ø        ENT  $
478Ø ;
479Ø ; DEMOSTRACION DE LAS
48ØØ ; RUTINAS
481Ø ; DEMO A
482Ø ;
483Ø        ENT  $
484Ø        LD   BC,(COORDX)
485Ø        LD   HL,(GRAFIC)
486Ø        CALL SPRITE
487Ø ;
488Ø ; LECTURA DEL TECLADO
489Ø ;
49ØØ TECLA  LD   A,253
491Ø        IN   A,(ØFE)
492Ø        BIT  Ø,A
493Ø        CALL Z,ABAJO
494Ø        LD   A,251
495Ø        IN   A,(ØFE)
496Ø        BIT  Ø,A
497Ø        CALL Z,ARRIBA
498Ø        LD   A,223
499Ø        IN   A,(ØFE)
5ØØØ        BIT  Ø,A
5Ø1Ø        CALL Z,DEREC
5Ø2Ø        LD   A,223
5Ø3Ø        IN   A,(ØFE)
5Ø4Ø        BIT  1,A
5Ø5Ø        CALL Z,IZQUI
5Ø6Ø        LD   A,127
5Ø7Ø        IN   A,(ØFE)
5Ø8Ø        BIT  Ø,A
5Ø9Ø        JR   Z,FUERA
51ØØ        LD   A,(PRES)
511Ø        CP   1
512Ø        CALL Z,PRESEN
513Ø        LD   A,Ø
514Ø        LD   (PRES),A
515Ø        JR   TECLA
516Ø ;
517Ø ; RUTINAS DEL MOVIMIENTO
518Ø ;
519Ø IZQUI  LD   A,(COORDX)
52ØØ        LD   HL,PASO
521Ø        SUB  (HL)
522Ø        LD   (COORDX),A
523Ø        CALL CAMBI
524Ø        LD   A,1
525Ø        LD   (PRES),A
526Ø        RET
527Ø ABAJO  LD   A,(COORDY)
528Ø        LD   HL,PASO
529Ø        SUB  (HL)
53ØØ        LD   (COORDY),A
531Ø        LD   A,1
532Ø        LD   (PRES),A
533Ø        RET
534Ø ARRIBA LD   A,(COORDY)
535Ø        LD   HL,PASO
536Ø        ADD  A,(HL)
537Ø        LD   (COORDY),A
538Ø        LD   A,1
539Ø        LD   (PRES),A
54ØØ        RET
541Ø DEREC  LD   A,(COORDX)
542Ø        LD   HL,PASO
543Ø        ADD  A,(HL)
544Ø        LD   (COORDX),A
545Ø        CALL CAMBD
546Ø        LD   A,1
547Ø        LD   (PRES),A
548Ø        RET
549Ø PRESEN LD   HL,(GRAFIC)
55ØØ        LD   BC,(COORDX)
551Ø        CALL MOVE
552Ø        RET
553Ø ;
554Ø ; VARIABLES DE LA DEMO
555Ø ;
556Ø COORDX DEFB 1ØØ
557Ø COORDY DEFB 1ØØ
558Ø PASO   DEFB 2
559Ø STATUS DEFB Ø
56ØØ GRAFIC DEFW GRAFI1
561Ø PRES   DEFB Ø
562Ø ;
563Ø ; RESTO DE LAS RUTINAS DE
564Ø ; LA DEMO
565Ø ;
566Ø FUERA  LD   HL,(GRAFIC)
567Ø        CALL SPRITB
568Ø        RET
569Ø CAMBD  LD   A,(STATUS)
57ØØ        CP   1
571Ø        RET  Z
572Ø        LD   A,1
573Ø        LD   (STATUS),A
574Ø        LD   HL,GRAFI2
575Ø        LD   (GRAFIC),HL
576Ø        LD   HL,GRAFI1
577Ø        CALL SPRITB
578Ø        LD   HL,GRAFI2
579Ø        LD   BC,(COORDX)
58ØØ        CALL SPRITE
581Ø        RET
582Ø CAMBI  LD   A,(STATUS)
583Ø        CP   Ø
584Ø        RET  Z
585Ø        LD   A,Ø
586Ø        LD   (STATUS),A
587Ø        LD   HL,GRAFI1
588Ø        LD   (GRAFIC),HL
589Ø        LD   HL,GRAFI2
59ØØ        CALL SPRITB
591Ø        LD   HL,GRAFI1
592Ø        LD   BC,(COORDX)
593Ø        CALL SPRITE
594Ø        RET
```

LISTADO 5

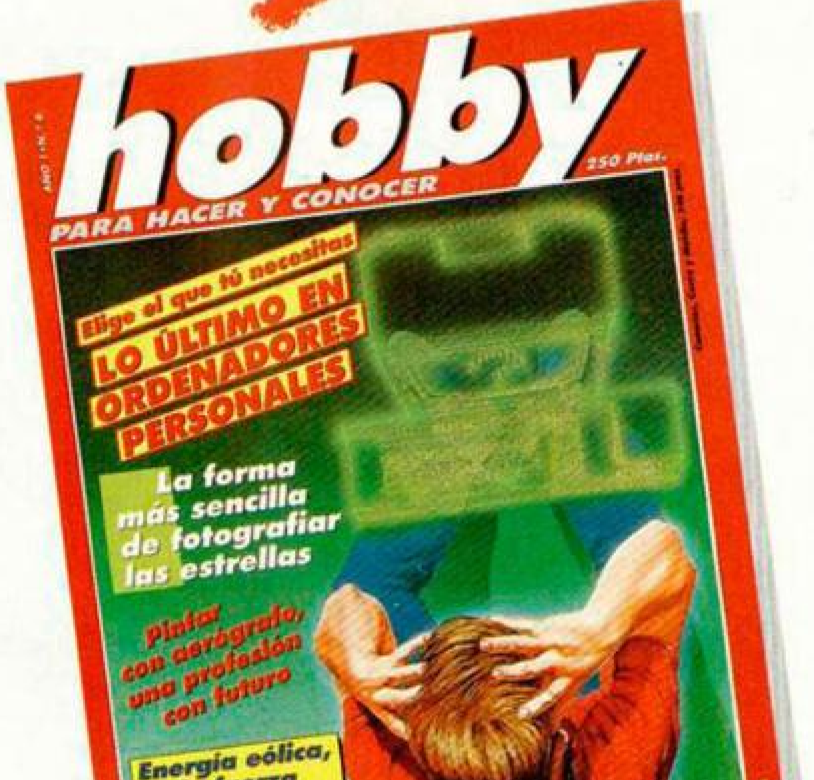
```
1   0000000000000000000000 0
2   000003F000000000000000 243
3   FC00000000000000070000 259
4   00000000003FFFFFC000 765
5   0000000400000040000000 128
6   009FFFC040000000000120 703
7   0040200000000002400080 290
8   2000000000480000802000 324
9   0000090000080100000000 153
10  120001001007E000006400 261
11  01001F820000008000200 300
12  0801003F0FFFFFE000801 605
13  3FC0000000000008014000 328
14  00003E0000014000000000 127
15  1C00007F407E0000000F 360
16  F001800007FFFE000001 886
17  800000000000000002400F 209
18  8000000780023030700 0 473
19  003870040FC008000040 451
20  080400400FFFFFC00FF8 1056
21  0040080000400800000020 176
22  1000002010000000201000 112
23  0020100000010200000010 112
24  2000000CC000000CC000 440
25  0003000000000300000000 6
26  0000000000000000000000 0
27  07F8000000000001FFF8 759
28  00000000000FFFF80000 518
29  0000003FFFFFE0000000 797
30  007FFFFFE00000000FF 1116
31  FFFFE00000000001FFFFFF 1500
32  E00000000003FFFFFFF000 1232
33  000007E001FFF0000000 727
34  0FC001FFF00000001F80 862
35  01FFF87E00007F0003FF 1015
36  FFFF0000FE0003FFFFFF 1532
37  003FFFFFFFFFFFFFFF3FFF 1911
38  FFFFFFFFFFFF7FFFFFFFF 2422
39  FFFFFFFFFFFFFFFFFFFFFF 2550
40  FFFFFFFFFFFFFFFFFFFFFF 2550
41  FFFFFFFFFFFFFFFFFFFFFF 2550
42  FFFFFFFFFFFFFFFFFFFFFF 2550
43  FFFFFFFFFFFFFFFFFFFFFF 2550
44  FFFF7FFFFFFFFFFFFFFFFE 2421
45  3FFFFFFFFFFFFFFFE0FFF 2117
46  FFFFFFFFFFFFC00FFFFFF 2292
47  FFFFFFFF8007FF800007F 1515
48  F800007FF800007FF800 998
49  003FF000003FF000001F 637
50  E000001FE000000FC000 686
51  000FC00000000300000003 213
52  0000000000000000000000 0
53  00000FC000000000000000 207
54  003F000000000000000000 63
55  E00000000003FFFFFC00 989
56  0000000020000002000000 4
57  000203FFF900000000004 513
58  0200004800000000040100 139
59  0240000000004010000120 104
60  0000000080100000900000 153
61  7E08000800004800000041F8 647
62  0080000260000080100040 374
63  00110000080100007FFFF0 783
64  FC0080100000000000003FC 651
65  8000007C000000002FE00 508
66  0038000000002800FF000 441
67  00007E028000007FFFE0 862
68  0001400000000000000001 66
69  4001E0000001F002200E 578
70  1C00000E0C0C20100200 116
71  001003F01FF003FFFFF0 1283
72  0200001002000000100200 38
73  0008040000008040000008 32
74  0400000804000000040800 28
75  0004080000003300000003 66
76  30000000C0000000C000 432
77  0000000000000000000000 0
78  1FE0000000000000001FFF 541
79  80000000000001FFFF000 654
80  00000007FFFFFC000000 769
81  0007FFFFFE0000000007 778
82  FFFFFFF00000000007FFFF 1282
83  FF8000000000FFFFFFFC0 1355
84  0000000FFF8007E00000 629
85  000FFF8003F000007E1F 798
86  FF8001F80000FFFFFFC0 1589
87  00FE0000FFFFFFC0007F 1338
88  0000FFFFFFFFFFFFFFC00 1782
89  FFFFFFFFFFFFFFFFFCFFFF 2547
90  FFFFFFFFFFFEFFFFFFFFFF 2549
91  FFFFFFFFFFFFFFFFFFFFFF 2550
92  FFFFFFFFFFFFFFFFFFFFFF 2550
93  FFFFFFFFFFFFFFFFFFFFFF 2550
94  FFFFFFFFFFFFFFFFFFFFFF 2550
95  FFFFFFFFF7FFFFFFFFFFFF 2422
96  FFFE7FFFFFFFFFFFFFFFFC 2418
97  3FFFFFFFFFFFFFFFF01FFF 2119
98  FFFFFFFFFFF00001FFE00 1560
99  001FFE00001FFE00001F 601
100 FE00000FFC000000FFC00 788
101 0007F8000007F8000003 513
102 F0000003F0000000C000 675
103 0000C00000000000000000 192
```



FIGURA 4

DUMP: 60000
BYTES: 1024

Esta es la mejor publicidad de hobby.



YA ESTÁ A LA VENTA EL N.º 6

RTTY

Mi pregunta es sobre el tema del RTTY abordado en el número 92 de la revista. El problema es que, si bien capto las ondas perfectamente, al conectar el Spectrum al receptor, éste se «come» la onda y la deforma, resultando el mensaje un tanto «chungo».

Quisiera que me dijerais qué puedo hacer, si hay algún aparato u otra cosa que lo evite. El receptor que uso es una radio de tres bandas.

Otra cosa: ¿Podríais explicarme qué es eso de entrada normal e invertida?

Andrés LOPEZ · **Cádiz (12)**

☐Como decíamos en nuestro artículo «TEXTOS POR EL ETER» del número 92, la conexión directa del ordenador a un rececptor del radio no da siempre resultados óptimos. No obstante, nos decidimos a publicar el programa para que, incluso aquellos que no pensaban adquirir el interface, pudieran experimentar en este terreno y no se quedaran «con la miel en los labios».

De momento, el programa le funciona perfectamente y la recepción y decodificación se lleva a cabo. El problema está, al parecer, en la pureza de las señales que está inyectando por la entrada «EAR». Para solucionar este problema, habrá que dejar un poco de lado la informática y hacer algo de experimentación electrónica.

Si la señal es débil, tal vez resultará útil un pequeño pre-amplificador. Aunque tiene el inconveniente de amplificar tanto señal como ruido. Más útil sería mejorar la relación señal/ruido instalando una buena antena. Lo ideal sería una antena de tejado, aunque si vive en un piso alto, puede probar a instalar un dipolo en la pared. Deberá tener forma de «V» invertida y bastante abierta. Cada rama tendrá una longitud igual a la cuarta parte de la longitud de onda que desee recibir. La conexión se hace en la parte alta de la «V» invertida, conectando el activo a una de las ramas y la masa a la otra.

En cualquier caso, los resultados dependen, en gran medida, de la calidad del receptor empleado. Pero no se desanime si no funciona a la primera. Lo más bonito de todas estas aficiones es, precisamente, la experimentación.

Respecto a la entrada normal o invertida, se trata de la posibilidad que tiene el programa de intercambiar las frecuencias de «MARK» y «SPACE» correspondientes al «1» y al «0» respectivamente, de cada carácter del código Baudot. A este respecto, puede encontrar amplia información en la página 23 del citado número 92, donde se explican las frecuencias de «MARK» y «SPACE» en cada una de las normas.

Amstrad PCW 8256

Soy propietario de un Spectrum y de un Amstrad PCW 8256 y, como supongo que ocurrirá a todo el que se encuentre en mi situación, me vendría muy bien utilizar la impresora del Amstrad para trabajar con el Spectrum. ¿Es esto posible?

F. Javier GOMEZ · **Madrid (11)**

☐Sin duda, lo peor del Amstrad PCW 8256 es su impresora, hay que tener en cuenta que se trata de una máquina que, a un precio muy reducido, incluye el que suele ser el accesorio más caro de un micro-ordenador. Esto trae consigo ciertos inconvenientes, el principal es que la impresora carece, por completo, de compatibilidad, ya que la circuitería necesaria para hacerla funcionar se encuentra dentro del propio ordenador y su conexión no sigue ningún protocolo normalizado. Por si fuera poco, el diseñar un interface para hacerla funcionar con otro ordenador, implicaría tener que diseñar toda la circuitería y software interno. Tenga en cuenta que la circuitería de una impresora es bastante similar a la de un ordenador, puesto que incluye un microprocesador, ROM y RAM. El trabajo sería similar al de diseñar un ordenador, por lo que con toda seguridad, no vale la pena.

Más interesante, en este sentido, sería poder transferir los datos desde el Spectrum al Amstrad, para que éste procediera a imprimirlos. Por lo que respecta al Spectrum, la conexión se podría hacer vía RS-232 mediante la salida del interface 1. En cuanto al Amstrad, existe una interface RS-232 disponible comercialmente, aunque nuestros compañeros de Amstrad Semanal podrían aconsejarle a este respecto con mayor seguridad.

Juegos en Basic

¿Cómo lograr, en un juego en Basic, que se borre sólo el muñeco y no toda la pantalla?

Cuando, en un juego en Basic nuevo, tengo que esperar a que mueva el enemigo para mover yo, ¿cómo se puede arreglar esto?

Alejandro GOETE · **Madrid (13)**

☐La forma de borrar un muñeco (por ejemplo, para cambiarlo de posición) es imprimir un espacio en el lugar que ocupa. Para ello, es necesario guardar en dos variables las coordenadas del muñeco en cada momento. Cada vez que haya que moverlo, se imprime un espacio en su lugar, se actualizan las coordenadas, y se vuelve a imprimir el muñeco en la nueva posición.

Respecto a su segunda pregunta, es una cuestión de planteamiento del programa. Si éste se desarrolla en un bucle en el que cada uno mueve una vez, su enemigo electrónico siempre tendrá ventaja, ya que él siempre moverá, mientras que el jugador humano es más fácil que se despiste. La solución es permitir que el jugador mueva varias veces por cada movimiento de la máquina. De esta forma, y si el bucle de juego es lo suficientemente rápido, parecerá que el jugador humano puede mover en todo momento.

Impresora GP-50S

Tengo una impresora GP-50S y me gustaría saber cómo puedo sacarle los distintos tipos de letras, ya que en el manual de instrucciones no viene nada explicado.

José A PEREZ · **Toledo (14)**

☐La GP-50S, a diferencia de las de 80 columnas, no puede trabajar con varios tipos de caracteres. La razón es que su funcionamiento no es autónomo, es el propio ordenador el que la hace funcionar, por lo que utiliza, exactamente, los caracteres que pueda imprimir en la pantalla del ordenador.

Efectivamente, se trata de una restricción importante, pero tenga en cuenta que es una impresora muy sencilla y de muy bajo precio.

Microfichas

¿Dónde puedo encontrar las microfichas M-0 y T-0?

¿Cómo puedo modificar

la rutina ON ERROR GOTO por una práctica ON ERROR CONTINUE?

Angel M. RODRIGUEZ - **Málaga (15)**

□ Las microfichas M-0 y T-0 no podrá encontrarlas en ninguna parte... por la sencilla razón de que no existe. Con el número «0» sólo se publicaron la G-0, I-0 y R-0 dado que contenían información necesaria para comprender las siguientes de su grupo. Sin embargo, en los grupos «T» y «M» esta información no era necesaria, por ello, las fichas «M-0» y «T-0» no existen.

No es cierto que una rutina ON ERROR CONTINUE sea más útil que una ON ERROR GOTO. Un error hay que procesarlo y corregirlo antes de que pueda continuar la ejecución del programa, por lo que se suelen colocar una serie de líneas que se encargan del procesamiento y corrección de los errores, antes de devolver el control a la secuencia principal.

Tenga en cuenta, no obstante, que los errores que procesan estas rutinas no son errores de programación, sino situaciones que son de error para el sistema, pero que han sido previstas por el programador y tienen una forma concreta de procesarse. Por ejemplo, un error de carga en un fichero que, en vez de detener el programa, hace aparecer un mensaje en la pantalla para que el usuario vuelva a intentar la carga. O la posibilidad de retornar al menú principal desde cualquier punto del programa pulsando la tecla «BREAK».

No obstante, si lo que desea es una rutina ON ERROR CONTINUE, no tiene más que hacerla saltar al comando siguiente a aquél donde se produjo el error, con lo que se resumirá la ejecución, pero el comando que ha producido el error no será ejecutado (si la ejecución se resumiese en el comando que ha producido error, se entraría en un bucle sin fin del que no sería posible salir).

Para ello, no tiene más que reentrar en el Basic sin actualizar los contenidos de «NEWPPC» y «NSPPC»; es decir, pasar directamente de la línea 290 a la 340 en el listado de la microficha R-1.

Código máquina

He intentado aprender código máquina con un libro. Al final intenté teclear este programa:

LD B, 0; 06h, 00h
LD C, 0; 0Eh, 00h
RET; C9h

Como no tengo ensamblador, tecleé los siguientes «POKEs»:

POKE 30000, 6
POKE 30001, 0
POKE 30002, 14
POKE 30003, 0
POKE 30004, 210

Al teclear: PRINT USR 30000 para que imprima el valor del registro doble «BC», el ordenador me hace un RESET y se me va todo al garete.

David SANCHEZ - **Madrid**

□ El error está en el código de RET que no es 210, sino 201. De todas formas, antes de teclear los «POKEs» haga: CLEAR 29999 para evitar problemas con el Basic.

DE OCASION

● ESTAMOS interesados en contactar con usuarios de Spectrum de toda España para intercambiar trucos, ideas, mapas, pokes, etc. Interesados llamar al tel. (983) 27 87 04. O bien escribir a José Antonio Santa Cruz. Puente Colgante, 11, 3.° D. 47007 Valladolid.

● NECESITO vender Spectrum Plus, con sus cables, fuente de alimentación, cinta de demostración, dos manuales y siete revistas. Además adjunto cassette Computone especial para ordenador y un TV Philips de 12" en b/n. Todo nuevo y en sus cajas correspondientes. Todo por tan sólo 35.000 ptas. Interesados en la compra pueden llamar al tel. (922) 21 82 53. Santa Cruz de Tenerife. Preguntar por Paco.

● VENDO teclado original de Spectrum por tan sólo 3.000 ptas. Alfredo. Tel. (911) 22 68 06.

● VENDO Spectrum 128 K español a estrenar. Incluyo lápiz óptico. Pon tú el precio. Preguntar por Andoni en horas de oficina llamando al tel. 443 98 89 de Bilbao.

● QUISIERA contactar con usuarios del Spectrum, a nivel nacional y poder formar un club. Interesados escribir a la siguiente dirección: José I. Ricarte. Puente Tables, 1-3, 1.° C. o bien llamar al tel. (976) 39 12 38. 50014 Zaragoza.

● VENDO Spectrum Plus, con todos los cables, transformador, cinta Horizontes, manual en inglés y castellano, joystick Quick Shot II e interface, cassette ideal para ordenador. Más de 40 revistas para el Spectrum. Libro «Aplicaciones para la casa y los pequeños negocios». Todo en buen estado y con su embalaje. Precio: 30.000 ptas. Interesados en la oferta pueden dirigirse a Gregorio Gacía. Tozal, 54. Fraga (Huesca).

● VENDO ordenador Toshiba T-350, unidad de disco 1-2 mb y disco duro 20 mb. Teclado en castellano, monitor, poco uso. Superoferta: 65.000 ptas. Interesados contactar con Ana, llamando al tel. 457 52 89 de 9 a 13 horas. Madrid.

● VENDO ordenador Zx Spectrum 48 K, teclado de goma, un poco más de un año de uso, tal y como fue comprado: joystick Quick Shot II más interface. Todo por 25.000 ptas. (convenir). Interesados escribir a José Gil Barrero Llano. Víctor Sáenz, 4, 6.° 1.ª. Oviedo (Asturias).

● VENDO ordenador Zx Sinclair ZX-81, completo, en perfecto estado por 4.000 ptas. También vendo revistas MICROHOBBY Semanal, ZX y Ordenador Popular. Interesados escribir a Carlos. Apartado 120. 03080 Alicante.

● OFERTA, vendo joystick Atari, sin estrenar, con interruptor de disparo a presión, sofisticada carcasa de plástico muy resistente. Precio 1.250 más gastos de envío. Interesados pueden escribir a Luis Espino Matas. Pedro Gimeno, 51, 1.° B. 03007 Alicante.

● DESEARIAMOS contactar con usuarios del Commodore y Spectrum. Si te interesa, llama al tel. (943) 21 09 76. Preguntar por Fernando.

● VENDO Spectrum 48 k, Saga I, más interface I, microdrive, interface para desproteger y pasar a microdrive, cassette especial para ordenador. Todo en perfecto estado, por 45.000 ptas. Barcelona. Tel. 224 90 94.

● DESEO intercambiar ideas, trucos, y todo lo que se relacione con el Spectrum Plus. Los interesados pueden llamar al tel. (948) 82 62 64 o bien escribir a José Julio Bocos García. P° Pamplona, 14, esc-7, 9.° B. Tudela (Navarra).

● VENDO Zx Spectrum 48 k, cinta de manejo y todos los accesorios para su funcionamiento, cassette especial ordenador, interface Dk'Tronics dos salidas, joystick Quick Shot II y 20 revistas. Todo por 30.000 ptas. Interesados llamar al tel. 228 12 71 de Madrid y preguntar por José, a partir de las 7 tarde.

● DESEARIA intercambiar pokes, trucos, mapas, etc. para Spectrum 48 k, sin interés económico. Escribir a Carlos García. Sol, 36, 4.° izda. 39003 Santander.

● VENDO Zx Spectrum, prácticamente nuevo, con fuente de alimentación, cables, cinta (Horizontes) y manuales en castellano, más interface y joystick (Kempston), fichero SITI. Precio a convenir. Dirigirse, cuanto antes, a Avd. Primavera, 5, 1.° 1ª. Cerdanyola del Vallés, Barcelona. O bien llamar al tel. 692 01 05 a partir de las 2 y hasta las 4 de la tarde. Preguntar por Adolfo.

● VENDO ordenador Spectrum 48 K, con manual y libro de instrucciones, cassette Horizontes, cables, fuente de alimentación y, además, regalo un interface tipo Kempston, más 25 revistas de MICROHOBBY. Para más información llamar al tel. 242 04 85 a partir de las 18 horas. Preguntar por Javier. Madrid.

● Vendo Zx Spectrum 48 K, teclado profesional Saga 1, fuente, joystick Quick Shot II, interface, cinta, manuales en inglés y español, libros, por sólo 23.000 ptas. También vendo impresora Star Gemini-10 (SG-10), interface paralelo, cables, 1.000 hojas de papel continuo, manuales. Precio 55.000 ptas. Interesados en estas ofertas pueden llamar al Tel. (956) 89 73 11 (Ext: 1160). Preguntar por Carlos.

● DESEARIA formar un club Spectrum. También deseo comprar números atrasados de la revista MICROHOBBY a 125 ptas. Interesados escribir a Javier Basilio. Mayor, 99, 10-D. Alcorcón (Madrid).

● SE HA formado un club de usuarios del Zx Spectrum en Murcia. Interesados en formar parte de éste, escribir a la siguiente dirección: José Angel de Castro Ortín. Geranio, 3. El Palmar (Murcia). Si quieres más información llama al tel. (968) 84 07 32.

● DESEARIA contactar con usuarios de Spectrum, preferiblemente de Barcelona, para intercambiar pokes, trucos o ideas. Asimismo desearía saber si existe algún club en Barcelona. Si es así, rogaría que algún miembro se pusiera en contacto conmigo. Llamar al tel. (93) 237 91 74 por las mañanas. Preguntar por Iván.

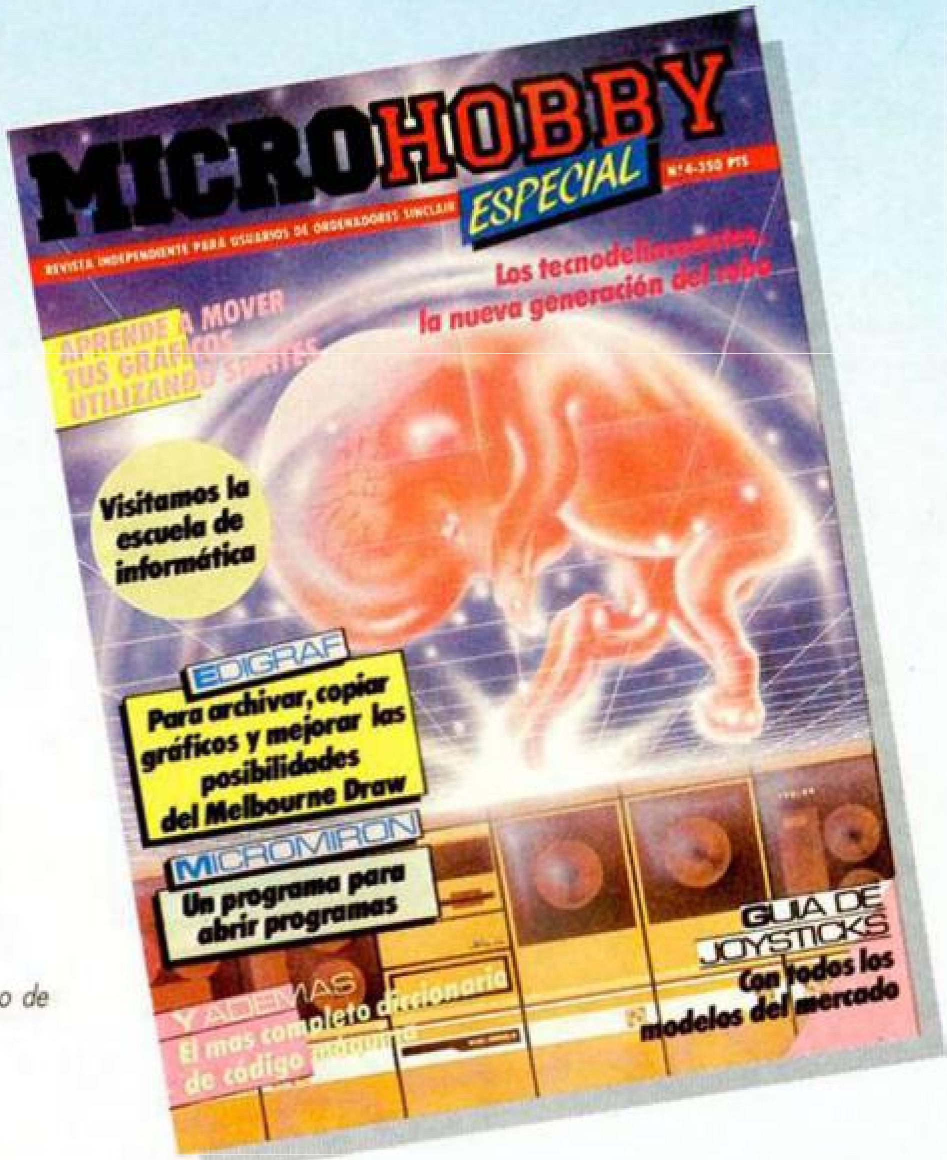
● VENDO Zx Spectrum Plus, con sus cables. Regalaría interface Kempston. Todo sólo 24.000 ptas. Telefonear al (986) 27 45 92.

● SE HA formado un club a nivel nacional, para usuarios de Spectrum, Commodore 64 y Amstrad o MSX. Tenemos toda clase de trucos, ideas, mapas, instrucciones de tus juegos. Prometemos contestar. Interesados escribir a Javier Jiménez. Valmojado, 221, 3.°. 28047 Madrid.

● VENDO Zx Spectrum 48 k con caja, fuente de alimentación, cables, manual, seis libros sobre programación del Spectrum, colección de fichas de lenguaje código máquina, colecciones completas de revistas. Todo por 35.000 ptas. Interesados en la compra dirigirse a Ramón Montero. Paseo Quince de Mayo, 12. 28019 Madrid. Tel. (91) 260 73 20.

● VENDO ordenador Zx Spectrum 48 K con cassette de demostración, instrucciones, con todo el embalaje y sus accesorios por 18.000 ptas. Interesados llamar al tel. 345 22 94 de Barcelona. A partir de las 7 y hasta las 10 de la noche. Preguntar por David.

El especial más especial de todos



Las técnicas más sofisticadas para mover gráficos por la pantalla explicadas paso a paso. Con un programa demostración y otro monitor que nos permite utilizarlo desde Basic.

José Gabriel Zato, Jefe de Estudios de la Escuela de Informática de Madrid, nos cuenta todo sobre el presente y el futuro de los estudios de ordenador en España.

Te ofrecemos la posibilidad de dotar al mejor programa de dibujo para Spectrum de nuevos comandos superpotentes para archivar cualquier figura, gráfico o zona de pantalla.

Líneas cero, códigos de control, Basic enmascarado, antimerge, nada se resiste a Micromiron, un programa capaz de abrir cualquier programa. Para que te conviertas en un auténtico Hacker.

El diccionario más completo de Código Máquina con todos los comandos del Z80 explicados uno a uno.

Contamos toda la verdad sobre una realidad cada vez más de actualidad, los ladrones informáticos. Quiénes son, cómo trabajan y en qué lugares suelen actuar.

Guía con los Joysticks que puedes encontrar ahora mismo en nuestro país. Incluye además un cuadro comparativo que explica las diferencias entre cada uno de ellos.

Si no lo encuentras en tu kiosco puedes solicitarlo directamente a nuestra editorial.

Recorta o copia este cupón y envíalo a Hobby Press, S. A. Apartado de Correos 232. Alcobendas (Madrid).

SI deseo recibir en mi domicilio el especial n.º **4** de Microhobby, al precio de 350 ptas.

Nombre .... Apellidos .... Fecha de nacimiento ....
Domicilio .... Localidad .... Provincia ....
C. Postal .... Teléfono ....

Forma de pago:
- ☐ Contra reembolso (supone 75 ptas. de gastos de envío).
- ☐ Mediante giro postal n.º ....
- ☐ Mediante tarjeta de crédito n.º ☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐
  Fecha de caducidad de la tarjeta ....
  ☐ Visa. ☐ Master Charge. ☐ American Express.
- ☐ Mediante talón nominativo a Hobby Press, S. A.

Fecha y firma

También puedes solicitarlo por teléfono: (91) 734 65 00.


DYNAMITE DAN II
El Heroe Vuelve!
Encuentra y destruye la planta secreta de prensaje de discos, escondida en el complejo de 8 islas de ARCANUM, 200 Pantallas llenas de astutos duendes, música y acción totalmente adictivas.
Spectrum
PRONTO: Amstrad
Amstrad Disk
ZAFI CHIP
La seguridad del Mundo esta en tus manos - podras eliminar al Doctor Blitzen otra vez?
MIRRORSOFT
ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141 28046 Madrid
Tel. 459 30 04 Telex 22690 ZAFIR E
Editado, fabricado y distribuido en España bajo la garantía Zafiro. Todos los derechos reservados.